GEOPAK 2001

TYPICAL SECTION GENERATOR HELP

The following appendices provide information to supplement the description files for each of the typical sections. It is a work in progress and will updated periodically. The "Home" button on the following pages will return you to this page.

Table of Contents

Appendix 1 Define Variables

File

These variables are used to define the name of the MicroStation file containing specific geometry used by the criteria files. **IMPORTANT**, if the file is not in the GEOPAK Project "Working Directory" then the complete drive letter and path **MUST** be specified. The plan element descriptions for each typical section state which elements need to be placed in which file. The following list gives all of the file variables and indicates which typical sections use a particular variable. Suggestions are also provided on what to include or exclude from a particular file.

Existing TOPO DGN

Used by Existing Features Typical Section.

Because many of the plan elements must cross the pattern line in pairs (edges of pavement, sidewalks, etc.) it is probably not a good idea to place all of the topographical information in this file. All that needs to be in this file are the items used to locate the existing features that are to be shown in the cross section.

Right of Way DGN

Used by Undivided New Pavement Typical Section

This drawing contains the existing and proposed right of way line(s).

Cross Section DGN

Used by Undivided New Pavement Typical Section

This drawing contains the existing cross sections.

Proposed Plan DGN

Used by Undivided New Pavement Typical Section

This drawing contains the proposed design elements.

GEOPAK Lines DGN

Used by Undivided New Pavement Typical Section

This drawing contains the proposed GEOPAK design elements that are normally shown on the plan sheets but are needed to control the criteria while processing the proposed cross sections. An example of an element shown in this file would be **Fill Slope 1 Breakline.** It is recommended to use the Pattern_Shape.DGN as the GEOPAK Lines DGN.

Plot Scale

The plot scale variables are used to define the plot scale of various drawings that have items place in them by the criteria. The value is the plot scale in the formula: 1”=Plot Scale’. Consequently a plot scale of 10 is used for a 1”=10’ scale sheet, 50 is used for a 1”=50’ scale sheet, etc. The Define Variables section of each typical section gives information on the use of its plot scales. The following list contains all of the plot scale variables and indicates which typical sections use which variables.

XS Scale

Used by Existing Features and Undivided New Pavement Typical Sections

This will be the scale of the cross section drawing. It is used in determining the text size for labeling slopes, elevations, and offsets on the proposed cross sections. For cross section plot scale of 1"=10' enter the number 10 as a value.

Plan View Scale

Used by Undivided New Pavement Typical Section

This will be the scale of the propsed plan drawing. It is used in determining the text size for labeling ditch slopes in the proposed plan view. For proposed plan plot scale of 1"=50' enter the number 50 as a value.

Plan View Flow Arrow Scale

Used by Undivided New Pavement Typical Section

This will be the scale of the proposed plan drawing ditch flow arrows. It is used in determining the size of the ditch flow arrow cell in the proposed plan view.

GPK Element

The GPK element variables are used to indicate the name of coordinate geometry objects that are used by the criteria. Typically they are used to draw to a particular location or elevation.

Survey Baseline Name

Used by Undivided New Pavement Typical Section

If a survey baseline is used that is different from the proposed baseline, the user may enter the name and the proposed cross sections will draw the baseline and label its offset. The default of "none" will cause this variable to be ignored. This alignment must be previously stored in COGO before processing cross sections.

Left Special Ditch Profiles & Right Special Ditch Profiles

Used by Undivided New Pavement Typical Section

These variables list the name(s) of the all of the optional **LEFT** & **RIGHT** special ditch profiles for a proposed cross section run. These profile(s) must be previously stored in COGO before processing cross sections. Commas must separate the profile names. Example: ltdit1,ltdit2,ltdit3 or rtdit1,rtdit2,rtdit3.

Left Ditch Alignment & Right Ditch Alignment

Used by Undivided New Pavement Typical Section

These variables list the name of the optional **LEFT** & **RIGHT** ditch alignments. It allows the user to control the location of the toe of the ditch resides, which is where the chain intersects the cross section. This is helpful when trying to match with an existing culvert for example. It overrides the ditch fore slope 2 variable and forces the ditch to have a varying foreslope. Only ONE alignment per side is allowed per cross section run. This alignment must be previously stored in COGO before processing the cross sections. Example: ltditch1 or rtditch1.

Left Match Line Profiles & Right Match Line Profiles

Used by Undivided New Pavement Typical Section

These variables list the name(s) of the optional **LEFT** & **RIGHT** match line profiles. A user may draw a graphical match line in the GEOPAK lines file and the cross section will stop drawing at that location and label it as a match line. This variable expands on that concept and allows the match line to also be controlled vertically. For example, a user draws a matchline through the fill slope 2 region. The fill slope 2 variable is set to a run:rise value of 3:-1. Without a match line profile, the fill slope 2 line would be drawn at a 3:-1 until intersecting the match line. This variable will force an "over ride" of the fill slope 2 variable and force the final fill slope 2 line to be drawn to the named match line profile. Commas must separate the profile names. Example: ml1lt,ml2lt,ml3lt or ml1rt,ml2rt,ml3rt.

Left Sidewalk Profiles & Right Sidewalk Profiles

Used by Undivided New Pavement Typical Section

These variables list the name(s) of the all of the optional **LEFT** & **RIGHT** independent sidewalk profiles for a proposed cross section run. This profile will be applied to the inside edge of the proposed sidewalk only when the sidewalk is not adjacent to the back of curb. These profile(s) must be previously stored in COGO before processing cross sections. The profile names are to be separated by commas. Example: ltsw1,ltsw2,ltsw3 or rtsw1,rtsw2,rtsw3

Left Interception Special Ditch Profiles & Right Interception Special Ditch Profiles

Used by Undivided New Pavement Typical Section

These variables list the name(s) of the all of the optional **LEFT** & **RIGHT** interception special ditch profiles located on the outside of the proposed levee. These profile(s) must be previously stored in COGO before processing cross sections. Commas must separate the profile names. Example: ltlev1,ltlev2,ltlev3 or rtlev1,rtlev2,rtlev3.

Median Ditch Profiles

Ramp Chain Names

Used by Undivided New Pavement Typical Section

This variable lists the name(s) of the all of the optional proposed ramp chain names. It is used for resolving paved gores. The ramp chain name spelling must exactly match the ramp profile name. For example, a user has a ramp chain named "ramp1". Then the profile name must also be "ramp1". The criteria will check to see if the pavement should continue from the edge of the mainline pavement to the ramp chain and profile. Commas must separate the profile names. Example: ramp1,ramp2,ramp3. **IMPORTANT:** The listing of the ramp chains MUST NOT include any blank spaces.

Match Line Chain Names

Used by Undivided New Pavement Typical Section

This variable lists the name(s) of the all of the optional proposed match line chain names. It is used for resolving grass gores. The match line chain name spelling must exactly match the match line profile name. For example, a user has a match line chain named "ML1". Then the profile name must also be "ML1". The criteria checks to see if the grass slopes are to draw to a matchline chain and profile. Commas must separate the profile names. Example: ML1,ML2,ML3. **IMPORTANT:** The listing of the match line chains MUST NOT include any blank spaces.

Left Top of Wall Profiles & Right Top of Wall Profiles

Used by Undivided New Pavement Typical Section

These variables list the name(s) of the all of the optional **LEFT** & **RIGHT** profiles to control the elevation of the top of retaining wall for a proposed cross section run. These profile(s) must be previously stored in COGO before processing cross sections. The profile names are to be separated by commas. Example: LTW1,LTW2,LTW3 or RTW1,RTW2,RTW3.

Left Top of Footing Profiles & Right Top of Footing Profiles

Used by Undivided New Pavement Typical Section

These variables list the name(s) of the all of the optional **LEFT** & **RIGHT** profiles to control the elevation of the top of the footing of retaining wall for a proposed cross section run. These profile(s) must be previously stored in COGO before processing cross sections. The profile names are to be separated by commas. Example: LTF1,LTF2,LTF3 or RTF1,RTF2,RTF3.

Median Top of Footing Profiles

Appendix 2 Redefinable Variables

Note: The text inside the parenthesis represents the variable name the user will see inside of Project Manager.

Show Redefinable Variables

The show redefinable variables control whether or not a specific feature will be shown in the cross section. Specify ^Yes^ or ^No^ for each feature listed. The carets "^" are required.

Show Existing Shoulders → (_s_ShowShoulders)

Used by Existing Features Typical Section

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_ShowShoulders = ^Yes^
    }
```

Show Existing Sidewalks → (_s_ShowSidewalk)

Used by Existing Features Typical Section

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_ShowSidewalk = ^Yes^
    }
```

Show Existing Paved Ditches → (_s_ShowPavedDitch)

Used by Existing Features Typical Section

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_ShowPavedDitch = ^Yes^
    }
```

Show Existing Paved Surfaces → (_s_ShowPavedSurface)

Used by Existing Features Typical Section

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_ShowPavedSurface = ^Yes^
    }
```

Show Existing Buildings → (_s_ShowBuildings)

Used by Existing Features Typical Section

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_ShowBuildings = ^Yes^
    }
```

Show Existing Railroad Tracks → (_s_ShowRailroadTracks)

Used by Existing Features Typical Section

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_ShowRailroadTracks = ^Yes^
    }
```

Show Existing Retaining Walls → (_s_ShowRetainingWall)

Used by Existing Features Typical Section

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_ShowRetainingWall = ^Yes^
    }
```

Show Existing Curb and Gutter → (_s_ShowCurbandGutter)

Used by Existing Features Typical Section

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_ShowCurbandGutter = ^Yes^
    }
```

Label Edge of Shoulder Elevations and Offsets → (_s_LabelShoulderElevations)

Used by Undivided New Pavement Typical Section

Controls whether to label edge of proposed shoulder elevations and offsets.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_LabelShoulderElevations = ^Yes^
    }
```

Existing Features Dimension Redefinable Variables

This set of variables controls the dimensions of existing features drawn in the cross sections. All values are in master units.

Existing Features Line Height → (_d_ExistingFeatureLineHeight)

Used by Existing Features Typical Section.

Defines the height of the line representing the location of the existing features. The number **MUST** be greater than zero.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_ExistingFeatureLineHeight = 10
    }
```

Existing Pavement Thickness → (_d_ExistPavtThick)

Used by Existing Features Typical Section.

This is the thickness of the existing pavement structure. The number **MUST** be greater than zero.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_ExistPavtThick = 1
    }
```

Existing Shoulder Thickness → (_d_ExistShouldThick)

Used by Existing Features Typical Section.

This is the thickness of the existing shoulder from the top of shoulder to the bottom. The number **MUST** be greater than zero.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _ _d_ExistShouldThick = 6/12
    }
```

Existing Sidewalk Thickness → (_d_ExistSidewalkThick)

Used by Existing Features Typical Section.

This is the total thickness of the existing sidewalk structure from the top of sidewalk to the bottom. The number **MUST** be greater than zero. See MoDOT Standard Plan 608.10.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_ExistSidewalkThick = 4/12
    }
```

Existing Paved Ditch Thickness → (_d_ExistPavedDitchThick)

Used by Existing Features Typical Section.

This is the total thickness of existing paved ditches from the top of existing ground to the bottom of the paved ditch. The number **MUST** be greater than zero. See MoDOT Standard Plan 609.15.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_ExistPavedDitchThick = 6/12
    }
```

Existing Paved Surface Thickness → (_d_ExistPavedSurfaceThick)

Used by Existing Features Typical Section.

This is the total thickness of existing paved surfaces from the top of existing ground to the bottom of the paved surface(s). The number **MUST** be greater than zero.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_ExistPavedSurfaceThick = 4/12
    }
```

Existing Curb and Gutter Thickness → (_d_ExistCurbGutterThick)

Used by Existing Features Typical Section.

This is the total thickness of existing curb and gutter from the top of existing surface to the bottom of the curb and gutter at the edge of gutter or existing pavement. The number **MUST** be greater than zero. See MoDOT Standard Plan 609.00.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_ExistCurbGutterThick = 7/12
    }
```

Existing Curb and Gutter Width → (_d_ExistCurbGutterWidth)

Used by Existing Features Typical Section.

This is the total width of the existing curb and gutter from the edge of existing pavement to the back of curb. The number **MUST** be greater than zero. See MoDOT Standard Plan 609.00.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_ExistCurbGutterWidth = 9/12
    }
```

Type Redefinable Variables

The type redefinable variables are used to select between different options within the criteria. All require a caret (^) before after the type, since these are string variables and GEOPAK uses carets to designate that a value is a text string rather than a numeric value.

Pavement Type → (_s_PavementType)

Used by Undivided New Pavement Typical Section

Proposed pavement type. Use ^C^ or ^B^ for concrete or bituminous pavement respectively. This will determine if the edge of pavement is closed vertically for concrete or 1:1 for bituminous.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_PavementType = ^B^
    }
```

Outside Shoulder Type → (_s_OutsideShoulderType)

Used by Undivided New Pavement Typical Section

Determines the material type for the "outside" or non-median shoulder. The options are as follows:
^C^ = Concrete
^B^ = Bituminous
^A^ = Aggregate
^E^ = Earth

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_OutsideShoulderType = ^B^
    }
```

Percent Slope Redefinable Variables

The percent slope redefinable variables are used to define the slope of a surface in percent. Enter the value as a percent without the percent sign (-2% slope is entered as –2). Include the negative sign when applicable.

Normal Pavement Slope → (_d_NormalPavementSlope)

Used by Undivided New Pavement Typical Section

Proposed pavement slope in percent. This is typically a -2 percent or similar. This value directly effects the way super elevation transitions are calculated. It represents normal pavement crown slope when no super elevation is present.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_NormalPavementSlope = -2
    }
```

Normal Outside Shoulder Slope → (_d_NormalOutsideShoulderSlope)

Used by Undivided New Pavement Typical Section

Proposed "outside" non-median shoulder slope in percent. This is typically a -4 percent or similar. It represents normal shoulder slope when no super elevation is present.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_NormalOutsideShoulderSlope = -4
    }
```

U2 Shoulder Slope → (_d_U2ShoulderSlope)

Used by Undivided New Pavement Typical Section

Defines proposed U2 shoulder slope in percent. This is typically a 2 percent or similar. It represents normal U2 shoulder slope.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_U2ShoulerSlope = 2
    }
```

Left Berm Slope → (_d_BermSlope_Left)

Used by Undivided New Pavement Typical Section

Proposed LEFT berm slope in percent. This is the earth slope adjacent to the proposed shoulder or curb.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_BermSlope_Left = 2
    }
```

Right Berm Slope → (_d_BermSlope_Right)

Used by Undivided New Pavement Typical Section

Proposed RIGHT berm slope in percent. This is the earth slope adjacent to the proposed shoulder or curb.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_BermSlope_Right = 2
    }
```

Sidewalk Cross Slope → (_d_SidewalkSlope)

Used by Undivided New Pavement Typical Section

Proposed sidewalk cross slope in percent format.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_SidewalkSlope = 2
    }
```

Maximum Allowable Type 1 Driveway Slope → (_d_Max_Type1_Driveway_Slope)

Used by Undivided New Pavement Typical Section

Proposed maximum type 1 driveway slope. If the driveway slope exceeds this value a message will be drawn alerting the user.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_Max_Type1_Driveway_Slope = 10
    }
```

Maximum Allowable Type 1 Driveway Roll Over Slope → (_d_Max_Type1_Roll_Over_Slope)

Used by Undivided New Pavement Typical Section

Proposed maximum type 1 driveway rollover slope. This variable checks the difference in slope between the entrance pad slope and tie down to ground slope. If this value is exceeded then a vertical curve entrance will be drawn.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_Max_Type1_Roll_Over_Slope = 10
    }
```

Maximum Allowable Type 2 Driveway Slope → (_d_Max_Type2_Driveway_Slope)

Used by Undivided New Pavement Typical Section

Proposed maximum type 2 driveway slope. If the driveway slope exceeds this value a message will be drawn alerting the user.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_Max_Type2_Driveway_Slope = 5
    }
```

Maximum Allowable Type 2 Driveway Roll Over Slope → (_d_Max_Type2_Roll_Over_Slope)

Used by Undivided New Pavement Typical Section

Proposed maximum type 2 driveway rollover slope. This variable checks the difference in slope between the entrance pad slope and tie down to ground slope. If this value is exceeded then a vertical curve entrance will be drawn.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_Max_Type2_Roll_Over_Slope = 10
    }
```

Maximum Allowable Type 3 Driveway Slope → (_d_Max_Type3_Driveway_Slope)

Used by Undivided New Pavement Typical Section

Proposed maximum type 3 driveway slope. If the driveway slope exceeds this value a message will be drawn alerting the user.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_Max_Type3_Driveway_Slope = 5
    }
```

Maximum Allowable Type 3 Driveway Roll Over Slope → (_d_Max_Type3_Roll_Over_Slope)

Used by Undivided New Pavement Typical Section

Proposed maximum type 3 driveway rollover slope. This variable checks the difference in slope between the entrance pad slope and tie down to ground slope. If this value is exceeded then a vertical curve entrance will be drawn.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_Max_Type3_Roll_Over_Slope = 10
    }
```

Maximum Allowable Type 4 Driveway Slope → (_d_Max_Type4_Driveway_Slope)

Used by Undivided New Pavement Typical Section

Proposed maximum type 4 driveway slope. If the driveway slope exceeds this value a message will be drawn alerting the user.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_Max_Type4_Driveway_Slope = 5
    }
```

Maximum Allowable Type 4 Driveway Roll Over Slope → (_d_Max_Type4_Roll_Over_Slope)

Used by Undivided New Pavement Typical Section

Proposed maximum type 4 driveway rollover slope in percent. This variable checks the difference in slope between the entrance pad slope and tie down to ground slope. If this value is exceeded then a vertical curve entrance will be drawn. Do not include the percent sign.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_Max_Type4_Roll_Over_Slope = 10
    }
```

Maximum Allowable Type 5 Driveway Slope → (_d_Max_Type5_Driveway_Slope)

Used by Undivided New Pavement Typical Section

Proposed maximum type 5 driveway slope. If the driveway slope exceeds this value a message will be drawn alerting the user.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_Max_Type5_Driveway_Slope = 5
    }
```

Maximum Allowable Type 5 Driveway Roll Over Slope → (_d_Max_Type5_Roll_Over_Slope)

Used by Undivided New Pavement Typical Section

Proposed maximum type 5 driveway rollover slope. This variable checks the difference in slope between the entrance pad slope and tie down to ground slope. If this value is exceeded then a vertical curve entrance will be drawn.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_Max_Type5_Roll_Over_Slope = 10
    }
```

Side Slope Redefinable Variables

Used by Undivided New Pavement Typical Section

The side slope redefinable variables are used to define the slope of a side slope in Run:Rise format. Example, 6:-1. Make sure to include the colon. Include the negative sign before the **Rise** when applicable. The Run must always be positive.

Left Ditch Fore Slope 1 → (_d_DitchForeSlope1_Left)

Optional LEFT ditch fore slope 1 value.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_DitchForeSlope1_Left = 6:-1
    }
```

Left Ditch Fore Slope 2 → (_d_DitchForeSlope2_Left)

Required LEFT ditch fore slope 2 value.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_DitchForeSlope2_Left = 6:-1
    }
```

Left Ditch Back Slope → (_d_DitchBackSlope_Left)

Required LEFT ditch back slope value.
Rise is ALWAYS a positive number.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_DitchBackSlope_Left = 3:1
    }
```

Right Ditch Fore Slope 1 → (_d_DitchForeSlope1_Right)

Optional RIGHT ditch fore slope 1 value.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_DitchForeSlope1_Right = 6:-1
    }
```

Right Ditch Fore Slope 2 → (_d_DitchForeSlope2_Right)

Required RIGHT ditch fore slope 2 value.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_DitchForeSlope2_Right = 6:-1
    }
```

Right Ditch Back Slope → (_d_DitchBackSlope_Right)

Required RIGHT ditch back slope value.
Rise is ALWAYS a positive number.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_DitchBackSlope_Right = 3:1
    }
```

Left Fill Slope 1 → (_d_FillSlope1_Left)

Optional LEFT Fill Slope 1 value.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _ _d_FillSlope1_Left = 6:-1
    }
```

Left Fill Slope 2 → (_d_FillSlope2_Left)

Required LEFT fill slope 2 value.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_FillSlope1_Right = 3:-1
    }
```

Right Fill Slope 1 → (_d_FillSlope1_Right)

Optional RIGHT Fill Slope 1 value.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_FillSlope1_Right = 6:-1
    }
```

Right Fill Slope 2 → (_d_FillSlope2_Right)

Required RIGHT fill slope 2 value.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_FillSlope1_Right = 3:-1
    }
```

Slope Control Redefinable Variables

The slope control redefinable variables are used to select between different options within the criteria. All require a caret (^) before after the variable value, since these are string variables and GEOPAK uses carets to designate that a value is a text string rather than a numeric value.

Aggregate Extension Slope → (_s_ExtensionSlope)

Used by Undivided New Pavement Typical Section

Determines whether aggregate base extension slope matches pavement slope or shoulder slope. Options are:
^P^ = pavement slope
^S^ = shoulder slope
This only has an effect when at the edge of pavement with aggregate base extension under concrete pavement.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_ExtensionSlope = ^S^
    }
```

Gutter Pan Slope In High Side Super Elevation → (_s_GutterSlopeInSuper)

Used by Undivided New Pavement Typical Section

Controls high side super gutter pan slope. Options are:
^U^ for up
^D^ for down
Effects type A and B curb and gutter only. If ^D^ or "down" is selected then the gutter will hold water on the high side of super elevation. If ^U^ or "up" is selected then the gutter pan will not hold water in the high side of super elevation.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_GutterSlopeInSuper = ^U^
    }
```

Force Closed Slope Control Redefinable Variables

Used by Undivided New Pavement Typical Section

The force closed slope control redefinable variables allow the user to force slope closure to a specific offset or at a slope other than the standard slope. Set the variable to ^Yes^ or ^No^ to force the slopes to close. When set to no, normal side slope conditions are used. When set to yes then a slope is drawn from the edge of the shoulder or curb directly to the ground. The carets "^" are required.

The closure can be controlled by either a plan view element or a forced slope variable. If a plan view element is drawn representing the location of the forced slope tie, then the plan element will take precedence and control the location of the tie. If the plan element is not found, variables are used to locate the tie. If the section is in cut the ForcedCutSlope value for that side is used and if the section is in fill the ForcedFillSlope value for that side is used. See the next page for information on the use of these two variables.

Left Force Closing Slopes → (_s_LeftForceClosingSlopes)

Used to force closing of LEFT side slopes.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_LeftForceClosingSlopes = ^No^
    }
```

Right Force Closing Slopes → (_s_RightForceClosingSlopes)

Used to force closing of RIGHT side slopes.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_RightForceClosingSlopes  = ^No^
    }
```

Force Closed Slope Value Redefinable Variables

Used by Undivided New Pavement Typical Section

The force closed slope value redefinable variables work in conjunction with the force closed slope control redefinable variables described on the previous page. If force closed slope control is set to ^Yes^ and a plan view element is not used to indicate the offset for the force sloped tie point, the slope used to close the slope is that specified by the appropriate force closed slope value. When set to no, normal side slope conditions are used. When set to yes then a slope is drawn from the edge of the shoulder or curb directly to the ground. The carets "^" are required.

The closure can be controlled by either a plan view element or a forced slope variable. If a plan view element is drawn representing the location of the forced slope tie, then the plan element will take precedence and control the location of the tie. If the section is in cut the ForcedCutSlope value for that side is used and if the section is in fill the ForcedFillSlope value for that side is used.

The forced slope value is in Run:Rise format. Run is ALWAYS a positive number. Rise is ALWAYS a positive number in CUT and ALWAYS a negative number in FILL (Example, 2:-1). Make sure to include the colon.

Left Forced Cut Slope → (_d_LeftForcedCutSlope)

Optional LEFT forced cut slope value.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_LeftForcedCutSlope = 2:1
    }
```

Left Forced Fill Slope → (_d_LeftForcedFillSlope)

Optional LEFT forced fill slope value.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_LeftForcedFillSlope = 2:-1
    }
```

Right Forced Cut Slope → (_d_RightForcedCutSlope)

Optional RIGHT forced cut slope value.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_LeftForcedCutSlope = 2:1
    }
```

Right Forced Fill Slope → (_d_RightForcedFillSlope)

Optional RIGHT forced fill slope value.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_RightForcedFillSlope = 2:-1
    }
```

Right of Way Constrained Closing Slope Redefinable Variables

Used by Undivided New Pavement Typical Section

The right of way constrained closing slope redefinable variables are used to force the slope to tie at the right of way line or inside of the right of way line with an optional buffer distance. If the normal slope can fit within the right of way then the normal slope will be used. If the normal slope will not fit within right of way, then a steeper slope will be drawn and labeled as such. This will allow the user to identify when their slopes were drawn steeper than the defaults.

Left Right of Way Constrained Closing Slope → (_s_LeftROWConstrainedSlope)

^Yes^ or ^No^ to force the LEFT slope to tie inside of the allowed right of way.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_LeftROWConstrainedSlope = ^No^
    }
```

Right Right of Way Constrained Closing Slope → (_s_LeftROWConstrainedSlope)

^Yes^ or ^No^ to force the RIGHT slope to tie inside of the allowed right of way.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_RightROWConstrainedSlope = ^No^
    }
```

Right of Way Buffer Width → (_d_ROW_Buffer_Width)

Provides a minimum buffer distance in master units when forcing the side slope tie down point inside of the right of way.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_ROW_Buffer_Width = 1
    }
```

Interception Ditch and Levee Redefinable Variables

Used by Undivided New Pavement Typical Section

The interception ditch and levee redefinable variables are used to draw an interception ditch and / or levee in cross section.

Levee Adjacent To Ditch Backslope → (_s_LeveeAdjacentToDitch)

^Yes^ or ^No^ to determine if optional levee is adjacent to the proposed ditch or if it is separated. If set to ^Yes^, the levee starting point will coincide with the tie point of the backslope of the proposed ditch. If set to ^No^, the levee will be drawn if top of levee inside and outside edges are found in the defined GEOPAK Lines DGN.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_LeveeAdjacentToDitch = ^No^
    }
```

Levee Height → (_d_LeveeHeight)

Defines the height of the optional levee in mater units as measured (1) from the existing ground if a interception ditch profile is not defined for the cross section station or (2) from the interception ditch profile if it is exists.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_LeveeHeight = 3
    }
```

Levee Fore Slope → (_d_LeveeForeslope)

Optional levee fore slope in Run:Rise format. Run and Rise are ALWAYS a positive number.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_LeveeForeslope = 2:1
    }
```

Levee Back Slope → (_d_LeveeBackslope)

Optional levee backslope value in Run:Rise format. Rise is ALWAYS a negative number.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_LeveeBackslope = 2:-1
    }
```

Levee Interception Ditch Width → (_d_InterceptionDitchWidth)

Optional levee interception ditch width in master units. This is a positive value and MUST be greater than zero.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_InterceptionDitchWidth = 2
    }
```

Levee Interception Ditch Backslope → (_d_InterceptionDitchBackslope)

Optional levee interception ditch backslope value in Run:Rise format. Rise is ALWAYS a positive number. Example, 2:1. Make sure to include the colon.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_InterceptionDitchBackslope = 2:1
    }
```

Rock Benching Redefinable Variables

Used by Undivided New Pavement Typical Section

The rock benching redefinable variables are used to draw side slope benches in cross section.

Rock Benching Backslope → (_d_BenchingBackSlope)

Rock benching backslope value in Run:Rise format.
Rise is ALWAYS a positive number.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_BenchingBackSlope = .1:1
    }
```

Rock Bench Width → (_d_BenchWidth)

Defines the tock bench width in master units.
This is a positive value and MUST be greater than zero

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_BenchWidth = 10
    }
```

Rock Bench Height → (_d_BenchHeight)

Defines the rock benching bench height in master units. Once this value is exceeded, a new bench will be drawn. This is a positive value and MUST be greater than zero.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_BenchHeight = 30
    }
```

Rock Top Bench Width → (_s_TopBenchWidth)

Defines the width of the top bench, which can be either to the line defined by the angle of repose or a specified width in master units. Set to ^RS^ for angle of repose slope or put in a number to specify a width --> i.e. ^10^. The carets "^" are required.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_TopBenchWidth = ^10^
    }
```

Pavement Thickness Redefinable Variables

Used by Undivided New Pavement Typical Section

The pavement thickness redefinable variables determine how the pavement is drawn. It is the distance from the top of the layer to the bottom of the layer. All values are in master units. The number for the first layer **MUST** be greater than zero. The number for all other layers may be greater than zero or zero but never less than zero. If it is set to zero that layer will not be drawn.

Pavement Layer 1 Thickness → (_d_PavementLayer1Thick)

This is the thickness of the first of four possible pavement structure thicknesses.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_PavementLayer1Thick = 1.75/12
    }
```

Pavement Layer 2 Thickness → (_d_PavementLayer2Thick)

This is the thickness of the second of four possible pavement structure thicknesses.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_PavementLayer2Thick = 0/12
    }
```

Pavement Layer 3 Thickness → (_d_PavementLayer3Thick)

This is the thickness of the third of four possible pavement structure thicknesses.

For an example see Pavement Layer 2 Thickness. The only difference is the name of the variable.

Pavement Layer 4 Thickness → (_d_PavementLayer4Thick)

This is the thickness of the fourth of four possible pavement structure thicknesses.

For an example see Pavement Layer 2 Thickness. The only difference is the name of the variable.

Shoulder Thickness Redefinable Variables

Used by Undivided New Pavement Typical Section

The shoulder thickness redefinable variables determine how the shoulder is drawn. It is the distance from the top of the layer to the bottom of the layer. All values are in master units. The number for the first layer **MUST** be greater than zero. The number for all other layers may be greater than zero or zero but never less than zero. If it is set to zero that layer will not be drawn. **Note**: If the thicknesses of the shoulder and the corresponding pavement layers match then no break will be drawn between the edge of pavement and the edge of shoulder. In essence, it will be one continuous layer of pavement spanning both the pavement and shoulder until a layer is reached for which the shoulder and pavement have different thicknesses.

Shoulder Layer 1 Thickness → (_d_ShoulderLayer1Thick)

This is the thickness of the first of four possible shoulder structure thicknesses.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_ShoulderLayer1Thick = 6/12
    }
```

Pavement Layer 2 Thickness → (_d_ShoulderLayer2Thick)

This is the thickness of the second of four possible shoulder structure thicknesses.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_ShoulderLayer2Thick = 0/12
    }
```

Shoulder Layer 3 Thickness → (_d_ShoulderLayer3Thick)

This is the thickness of the third of four possible shoulder structure thicknesses.

Shoulder Layer 4 Thickness → (_d_ShoulderLayer4Thick)

This is the thickness of the fourth of four possible shoulder structure thicknesses.

U2 Shoulder Thickness Redefinable Variables

Used by Undivided New Pavement Typical Section

The U2 shoulder thickness redefinable variables determine how the U2 shoulder is drawn. It is the distance from the top to the bottom of the layer. All values are in master units. The number **MUST** be greater than zero or equal to zero. If it is set to zero the layer will not be drawn. See PDM Figure 6-04.1 for suggested values.

U2 Shoulder Thickness → (_d_U2ShoulderThickness)

This is the proposed U2 shoulder thickness.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_U2ShoulderThickness = 2/12
    }
```

U2 Shoulder Aggbase Thickness → (_d_U2ShoulderAggbaseThickness)

This is the proposed U2 shoulder aggregate base thickness.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_U2ShoulderAggbaseThickness = 4/12
    }
```

Roadside Features Thickness Redefinable Variables

Used by Undivided New Pavement Typical Section

The roadside features thickness redefinable variables determine the thickness for various non-roadway items. All values are in positive. This number **MUST** be greater than zero.

Sidewalk Thickness → (_d_SidewalkThickness)

Provides the thickness of the proposed sidewalk.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_SidewalkThickness = 4/12
    }
```

Base Layer Thickness Redefinable Variables

Used by Undivided New Pavement Typical Section

The base layer thickness redefinable variables determine how the pavement base is drawn. All values are in master units, which can be zero or greater than zero, except as noted below. If it is set to zero the layer will not be drawn.

Aggregate Base 1 Thickness → (_d_Aggbase1Thickness)

Proposed aggregate pavement base 1 thickness. This number **MUST** be greater than zero if a thickness is specified for optional Aggbase 2 thickness. If an Aggbase 2 is thickness is greater than zero, this layer will be drawn as a Stabilized Permeable Base.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_Aggbase1Thickness = 4/12
    }
```

Aggregate Base 2 Thickness → (_d_Aggbase2Thickness)

Proposed aggregate pavement base 2 thickness. If greater than zero, Aggbase 1 will be drawn as a Stabilized Permeable Base.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_Aggbase2Thickness = 0/12
    }
```

Rock Fill Base Thickness → (_d_RockFillBaseThickness)

Proposed rock fill base thickness.

For an example see Aggregate Base Layer 2 Thickness. The only difference is the name of the variable.

Compensating Depth → (_d_CompensatingDepth)

Proposed compensating depth measured from the top of the proposed pavement.

For an example see Aggregate Base Layer 2 Thickness. The only difference is the name of the variable.

Rock Fill Base Daylighting Redefinable Variables

Used by Undivided New Pavement Typical Section

The rock fill base daylighting redefinable variables control the extension of the rock fill base from under the pavement. These options are given to conform to MoDOT practice as specified in the Project Development Manual (PDM) section 6-03.7:

Rock base consists of 18 in. [0.45 m] of Class C excavation. It is placed full roadbed width and daylighted to the in-slope or fillslope, except on the high side of superelevated curves and when not economically feasible under light duty pavements, in which case the rock base is not daylighted to the ditch and soil is used as a fill material at these locations (Rev. 10-10-03).

The variable value can be either ^Yes^ or ^No^ to "daylight" the rock fill base. The carets "^" are required.

The examples below give the default values for the variables.

Rock Fill Base Daylight → (_s_RockFillBaseDaylight)

Controls whether or not to "daylight" the rock fill base. If the criteria determines that it is not possible to daylight the rock fill base then it will close off the rock fill base at a 1:1 slope beyond the shoulder or curb and gutter.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
   {
      _s_RockFillBaseDaylight = ^Yes^
   }
```

Rock Fill Base Daylight High Side Super Elevation → (_s_RockFillBaseDaylightHSS)

Controls whether or not to "daylight" the rock fill base in the high side of super elevation (HSS). This will be an "override" of the variable _s_RockFillBaseDaylight for high side super elevation only.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
   {
      _s_RockFillBaseDaylightHSS = ^No^
   }
```

Ditch Dimension Redefinable Variables

Used by Undivided New Pavement Typical Section

The ditch dimension redefinable variables control the way ditches are drawn in cross section. Values are in master units. Depths are considered positive and must be greater than zero. Widths may be zero for a "V" ditch or greater than zero for a flat bottom ditch.

Left Standard Ditch Depth → (_d_StandardDitchDepth_Left)

Controls depth of LEFT standard ditch. This value is measured from the edge of shoulder, back of curb or edge of berm.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_StandardDitchDepth_Left = 4
    }
```

Right Standard Ditch Depth → (_d_StandardDitchDepth_Right)

Controls depth of RIGHT standard ditch. This value is measured from the edge of shoulder, back of curb or edge of berm.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_StandardDitchDepth_Right = 4
    }
```

Left Standard Ditch Width → (_d_DitchWidth_Left)

Controls width of LEFT standard ditch.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_DitchWidth_Left = 8
    }
```

Right Standard Ditch Width → (_d_DitchWidth_Right)

Controls width of RIGHT standard ditch.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_DitchWidth_Right = 8
    }
```

Ditch Text String Redefinable Variables

Used by Undivided New Pavement Typical Section

The ditch text string redefinable variables control miscellaneous ditch setting using text string values. Values are place between carets "^", which are required.

Special Ditch Label → (_s_SpecialDitchLabel)

Provides the content of special ditch labels i.e. ^S.D.^. This will be drawn on all ditches that use a special ditch profile.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_SpecialDitchLabel = ^S.D.^
    }
```

Draw Ditches In Plan View → (_s_DrawDitchesInPlanView)

Set to ^Yes^ or ^No^ to draw ditches and flow arrows in plan view while processing proposed cross sections.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_DrawDitchesInPlanView = ^No^
    }
```

Draw Underdrain Redefinable Variables

Used by Undivided New Pavement Typical Section

The draw underdrain redefinable variables control whether or not to include edge underdrains. These options are given to conform to MoDOT practice as specified in the Project Development Manual (PDM) section 6-02.3:

Pavement edge drains are required for new rigid or flexible pavements on medium and heavy duty routes, and permeable base courses will be provided on all heavy duty pavements with the following exceptions. Pavement edge drains and permeable base courses are not required where a minimum of 2 ft. [0.6 m] of daylighted rock base can be furnished for the top of the subgrade or where hydraulically placed sand fill comprises the top 4 ft. [1.2 m] of the embankment with not more than 2 ft. [0.6 m] of soil cap on the slopes. Thin courses of permeable bases cannot be relied upon to provide permanent drainage when daylighted and should never be used without longitudinal edge drains (Rev. 7-1-98).

Separate variables are given for the left and right side. The variable value can be either ^Yes^ or ^No^ to draw the underdrain under the proposed pavement. When set to ^Yes^ the underdrain will be drawn for that side except for the following two conditions. Underdrain will NOT be drawn if the pavement slope is greater than zero. In other words the pavement cross slope does not flow towards the shoulder. Secondly, an underdrain will not be drawn if the rock fill base is present and set to daylight. The carets "^" are required.

The examples below give the default values for the variables.

Draw Left Underdrain → (_s_DrawLeftUnderdrain)

Controls the left side.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_DrawLeftUnderdrain = ^No^
    }
```

Draw Right Underdrain → (_s_DrawRightUnderdrain)

Controls the right side.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_DrawRightUnderdrain = ^No^
    }
```

Underdrain Dimension Redefinable Variables

Used by Undivided New Pavement Typical Section

The underdrain dimension redefinable variables control the size of the underdrains shown in cross section. The values must be in positive master units. See MoDOT Standard Plan 605.10

Underdrain Height → (_d_UnderdrainHeight)

Proposed underdrain height as measured from the bottom of the pavement.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_UnderdrainHeight = 12/12
    }
```

Underdrain Width → (_d_UnderdrainWidth)

Proposed underdrain width.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_UnderdrainWidth = 1
    }
```

Curb Dimension Redefinable Variables

Used by Undivided New Pavement Typical Section

The curb dimension redefinable variable controls the size of the curb shown in cross section. The values must be in positive master units. See MoDOT Standard Plan 609.00

Integral Curb Height → (_d_IntegralCurbHeight)

Proposed curb height for type A, B, & S curbs only. It is the distance up from the gutter line.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_IntegralCurbHeight = 6/12
    }
```

Nonstandard Curb & Gutter Dimension Redefinable Variables

Used by Undivided New Pavement Typical Section

Important: Theses curb dimension redefinable variable are used only for nonstandard curb and gutters that do not conform to MoDOT Standard Plan 609.00. They are not to be used on state routes and are only provided to allow for cross sections to be drawn using municipal and other entity's standards.

Top of Gutter Slope → (_d_GutterSlope)

Slope in percent format of the top of gutter. Do not include the percent sign.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_GutterSlope = -5.2
    }
```

Gutter Width → (_d_GutterWidth)

Total width of curb and gutter in master units.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_GutterWidth = 2
    }
```

Curb Face Width → (_d_CurbFaceWidth)

The horizontal width of curb face in master units as measured from the gutter line to the start of the top of the curb.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_CurbFaceWidth = 2/12
    }
```

Top of Curb Width → (_d_TopCurbWidth)

The width of the top of the curb in master units as measured from the face to the back of the curb.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_TopCurbWidth = 6/12
    }
```

Curb Height → (_d_CurbHeight)

The height of the curb in master units as measured from the gutter line to the top of the curb.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_CurbHeight = 6.17/12
    }
```

Gutter Thickness → (_d_GutterThickness)

The thickness of the gutter in master units measured at the roadway face of the gutter.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_CurbHeight = 6.17/12
    }
```

Bottom of Gutter Slope → (_d_GutterBaseSlope)

Slope in percent format of the bottom of the gutter. Do not include the percent sign.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_GutterSlope = -5.2
    }
```

Entrance Dimension Redefinable Variables

Used by Undivided New Pavement Typical Section

The entrance dimension redefinable variables control the way entrances are drawn in cross section. The values must be in positive master units greater than zero. See MoDOT Standard Plans 203.61-203.65.

Entrance Pad Width → (_d_EntrancePadWidth)

Defines the distance from the edge of pavement to the entrance break point. If less than shoulder width then shoulder width will be used.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_EntrancePadWidth = 10
    }
```

Entrance Vertical Curve Length → (_d_EntranceVertCurveLength)

Defines the length of the vertical curve used for entrances.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_EntranceVertCurveLength = 20
    }
```

Retaining Wall Dimension Redefinable Variables

Used by Undivided New Pavement Typical Section

The retaining wall dimension redefinable variables control the way walls are drawn in cross section. The values must be in positive master units greater than zero unless noted otherwise.

Retaining Wall Depth Below Proposed Ground In Cut → (_d_DepthBelowProposedGroundInCut)

Defines the distance below proposed surface to top of retaining wall footing in a cut wall.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_DepthBelowProposedGroundInCut = 2
    }
```

Retaining Wall Height Above Existing Ground In Cut → (_d_HeightAboveExistGroundInCut)

Defines the distance from existing ground to the top of the retaining wall in a cut wall.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_HeightAboveExistGroundInCut = 0
    }
```

Retaining Wall Depth Below Existing Ground In Fill → (_d_DepthBelowExistGroundInFill)

Defines the distance below existing ground surface to top of retaining wall footing in a fill wall.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_DepthBelowExistGroundInFill = 2
    }
```

Retaining Wall Height Above Proposed Ground In Fill → (_d_HeightAbovePropGroundInFill)

Defines the distance from the proposed ground to the top of the retaining wall in a fill wall.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_TotalFootingWidth = 7
    }
```

Retaining Wall Width → (_d_WallWidth)

Defines the width of the retaining wall.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_WallWidth = 1
    }
```

Retaining Wall Height → (_s_WallHeight)

The user may decide to use the existing and proposed surfaces OR set a fixed height. The following options are available:
^GS^ = Use existing and proposed ground surfaces.
A Number equals actual wall height i.e. ^10^ for a 10 foot high wall. ALWAYS use carets.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _s_WallHeight = ^GS^
    }
```

Retaining Wall Back Wall Height In Cut → (_d_BackWallHeightInCut)

Defines the height of the retaining wall footing back wall. Use positive number.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_BackWallHeightInCut = 0.5
    }
```

Retaining Wall Back Footing Width → (_d_BackFootingWidth)

Defines the width footing behind the wall (distance from the back of the wall to the footing heel.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_BackFootingWidth = 4.5
    }
```

Retaining Wall Total Footing Width→ (_d_TotalFootingWidth)

Defines the total width of the retaining wall footing (distance from the footing toe to the heel.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_TotalFootingWidth = 7
    }
```

Retaining Wall Footing Thickness → (_d_FootingThickness)

Defines the thickness of the retaining wall footing.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_FootingThickness = 1.5
    }
```

Retaining Wall Key Offset → (_d_KeyOffset)

Defines the retaining wall footing key offset (Distance from the footing toe to the front of the key).

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_KeyOffset = 1.5
    }
```

Retaining Wall Key Width → (_d_KeyWidth)

Defines the width of the retaining wall footing key.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_KeyWidth = 1
    }
```

Retaining Wall Key Height → (_d_KeyHeight)

Defines the height of the retaining wall footing key.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_KeyHeight = 2
    }
```

Default Width Redefinable Variables

Used by Undivided New Pavement Typical Section

The default width redefinable variables are used only when no plan elements are found to determine the width of a cross section item. The value is in master units and can be zero or greater than zero. If the value is zero and the corresponding plan element is not found, the item will not be drawn. A plan element will **ALWAYS** "override" these variables.

Outside Shoulder Width → (_d_OutsideShoulderWidth)

Proposed shoulder width of the "outside" or non-median shoulder when no plan element is found designating the width.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_OutsideShoulderWidth = 8
    }
```

U2 Shoulder Width → (_d_U2ShoulderWidth)

U2 shoulder width when no plan element is found designating the width. This is measured from the back of curb.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_U2ShoulderWidth = 0
    }
```

Left Berm Width → (_d_BermWidth_Left)

LEFT berm slope width when no plan element is found designating the width. This is measured from the back of curb or edge of shoulder. **IMPORTANT:** In the event a sidewalk is required, the berm width must be set such that the back edge of the berm must be beyond the back of the sidewalk or the sidewalk will not be drawn.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_BermWidth_Left = 0
    }
```

Right Berm Width → (_d_BermWidth_Right)

RIGHT berm slope width when no plan element is found designating the width. This is measured from the back of curb or edge of shoulder. **IMPORTANT:** In the event a sidewalk is required, the berm width must be set such that the back edge of the berm must be beyond the back of the sidewalk or the sidewalk will not be drawn.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_BermWidth_Left = 0
    }
```

Left Ditch Fore Slope 1 Width → (_d_DitchForeSlope1Width_Left)

LEFT ditch slope 1 width when no plan element is found designating the width. This is measured from the back of curb or edge of shoulder or edge of berm.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_DitchForeSlope1Width_Left = 0
    }
```

Right Ditch Fore Slope 1 Width → (_d_DitchForeSlope1Width_Right)

RIGHT ditch slope 1 width when no plan element is found designating the width. This is measured from the back of curb or edge of shoulder or edge of berm.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_DitchForeSlope1Width_Right = 0
    }
```

Left Fill Slope 1 Width → (_d_FillSlope1Width_Left)

LEFT optional fill slope 1 width when no plan element is found designating the width. This is measured from the back of curb, edge of shoulder or edge of berm. If set to zero, and no plan element found then no fill slope 1 will be drawn.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_FillSlope1Width_Left = 24
    }
```

Right Fill Slope 1 Width → (_d_FillSlope1Width_Right)

RIGHT optional fill slope 1 width when no plan element is found designating the width. This is measured from the back of curb, edge of shoulder or edge of berm. If set to zero, and no plan element found then no fill slope 1 will be drawn.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_FillSlope1Width_Right = 24
    }
```

Search Distance Redefinable Variables

The search distance redefinable variables determine how far to search in the plan view for certain items. If the item is located within the search distance, it will be included in the cross section criteria process. If the item is beyond the search distance it will not be used by the criteria. The variables are required to avoid locating a plan view element from an adjacent or parallel roadway. All values are in master units. The number **MUST** be greater than zero.

Outside Shoulder Search Distance → (_d_OutsideShoulderSearchDistance)

Used by Undivided New Pavement Typical Section

Used to locate an "outside" non-median shoulder line. The distance is measured from the proposed edge of pavement. This variable is required to avoid locating a plan view shoulder element from an adjacent or parallel roadway.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_OutsideShoulderSearchDistance = 18
    }
```

Curb Search Distance → (_d_CurbSearchDistance)

Used by Existing Features Typical Section.

Used to locate a curb and or curb & gutter or gutter line. The distance is measured from the edge of pavement or shoulder depending on location. For existing curb, it is always measure from the edge of existing pavement.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_CurbSearchDistance = 3
    }
```

U2 Shoulder Search Distance → (_d_U2ShoulderSearchDistance)

Used by Undivided New Pavement Typical Section

Allowable search distance in feet to locate an "outside" edge line of the U2 shoulder located behind the proposed curb and or curb and gutter. This distance is measured from the proposed back of curb.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_U2ShoulderSearchDistance = 10
    }
```

Ramp Chain Search Distance → (_d_RampChainSearchDistance)

Used by Undivided New Pavement Typical Section

Defines the proposed ramp(s) search distance to locate the ramp chain(s) used for resolving paved gores. This distance is measured from the edge of the "mainline" pavement.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_RampChainSearchDistance = 25
    }
```

Match Line Chain Search Distance → (_d_MatchLineChainSearchDistance)

Used by Undivided New Pavement Typical Section

Distance measured from the edge of shoulder, curb, or berm to look for a match line chain. Primarily used for grass gore resolution.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_MatchLineChainSearchDistance = 100
    }
```

Retaining Wall Adjacent Wall Search Distance → (_d_AdjacentWallSearchDistance)

Used by Undivided New Pavement Typical Section

This variable is used to determine if the retaining wall is adjacent to the shoulder or not. If the wall is closer to the edge of shoulder than this distance, then the aggbase(s) will draw directly to the face of the wall.

EXAMPLE:

```
if (Sta >= 0+00 R 1)
    {
        _d_AdjacentWallSearchDistance = 12/12
    }
```

Appendix 3 Existing Features

Roadway

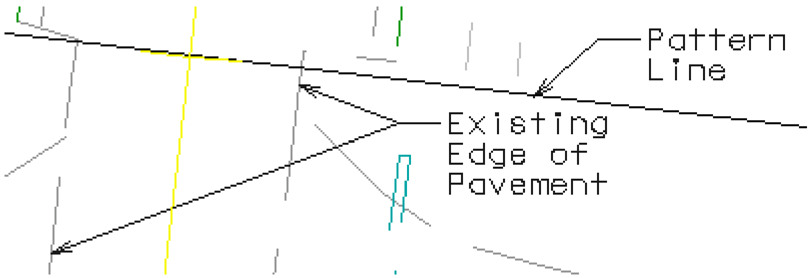
Existing Pavement

Drawn by Existing Features Typical Section.

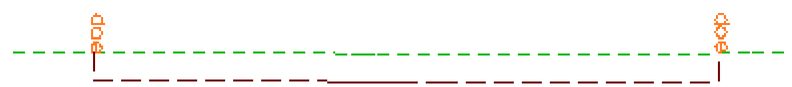
For the existing pavement to be drawn in the cross section, an Existing Edge of Pavement line must be drawn in the designated Existing TOPO DGN. The symbology for the line is: Level 20; Color 112; Style dash 3; and Weight 2. Both edges of the pavement must cross the pattern line for it to be drawn correctly. It is recommend that any existing edges of pavement that are not to be used for showing existing pavement in the cross section view be placed in a DGN other than the one defined as the Existing TOPO DGN.

The thickness (in master units) shown in the cross-section is controlled by the redefinable variable **_d_ExistPavtThick**.

Example



Existing TOPO DGN view



Cross-Section View

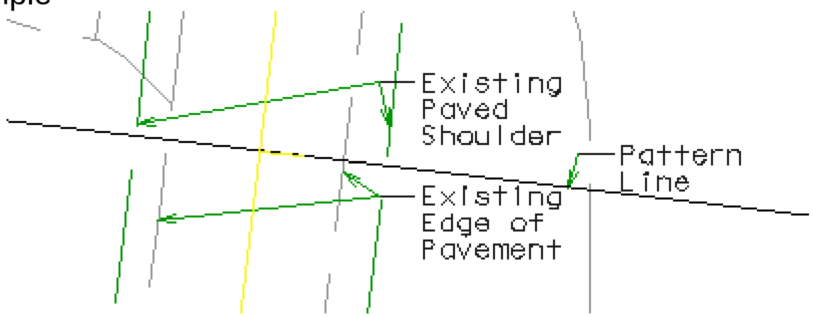
Existing Paved Shoulder

Drawn by Existing Features Typical Section.

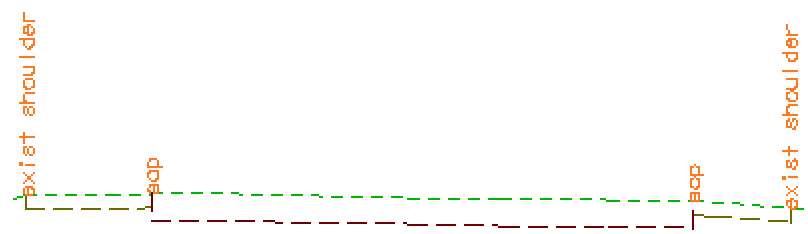
For the existing paved shoulder to be drawn in the cross section, an Existing Paved Shoulder line must be drawn in the designated Existing TOPO DGN at the outside edge of the shoulder and the redefinable variable **_s_ShowShoulders** must be set to ^yes^ for the current station. The symbology for the line is: Level 18; Color 114; Style dash 3; Weight 2. It must be on the outside of the corresponding Existing Edge of Pavement line since the paved shoulder will be drawn from the existing edge of pavement to the existing edge of shoulder as shown the figures below.

The thickness (in master units) shown in the cross-section is controlled by the redefinable **_d_ExistShouldThick**.

Example



Topo View



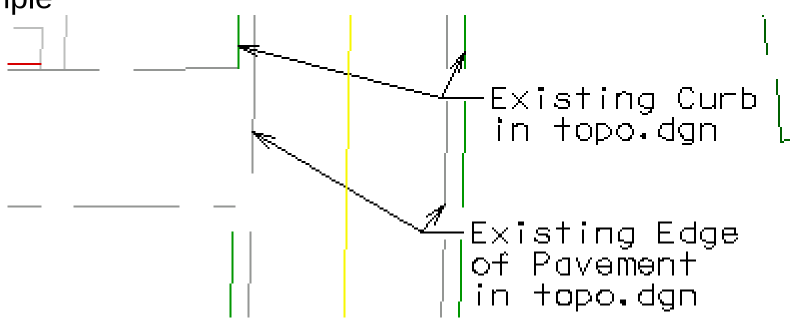
Cross-Section View

Existing Curb and Gutter

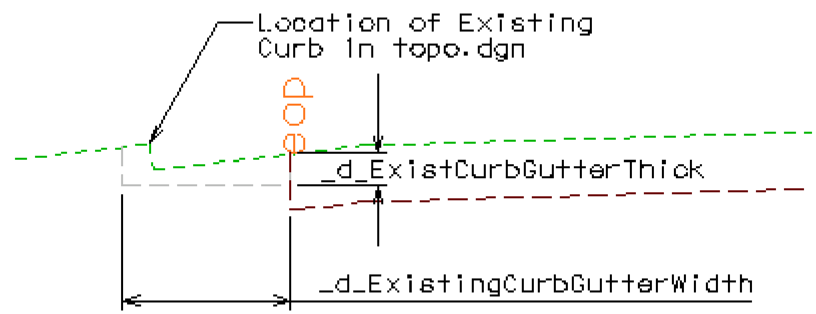
Drawn by Existing Features Typical Section.

For the existing curb to be drawn in the cross section, an Existing Curb line (Level 19; Color 114; Style dash 3; Weight 2) must be drawn in the designated Existing TOPO DGN outside of an Existing Edge of Pavement line within the allowable search distance set by **_d_CurbSearchDistance**. In addition, the redefinable variable **_s_ShowCurbandGutter** must be set to ^yes^ for the current station. The Existing Curb line only indicates the presence of an existing curb. It **does not** indicate the offset of the curb. Rather, the existing curb is drawn using the following parameters: The Existing Edge of Pavement line determines the inside edge of the curb or curb & gutter (this point is marked with the text 'eop' in the cross section views in the figures shown below); **_d_ExistCurbGutterThick** determines the depth of the inside face, as measured down from the top of the existing surface (see the upper right cross section view); **_d_ExistCurbGutterWidth** locates the back of the curb as measured out from the edge of pavement (see the upper right cross section view); a vertical line is drawn from that point to existing ground. In most cases **the user will need to manually place the existing edge of pavement line** at the proper location for the criteria to work. If the existing curb line already exists in the drawing, the easiest way to place the existing edge of pavement line is to offset the existing curb line by the appropriate distance using the MicroStation **Copy Parallel** tool with the Existing Edge of Pavement line selected in **D&C manager** and **Place Influence** activated.

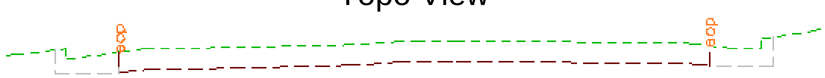
Example



Topo View



Cross-Section View



_d_ExistingCurbGutterThick
= _d_ ExistPavedDitchThick

_d_ExistingCurbGutterThick
< _d_ ExistPavedDitchThick

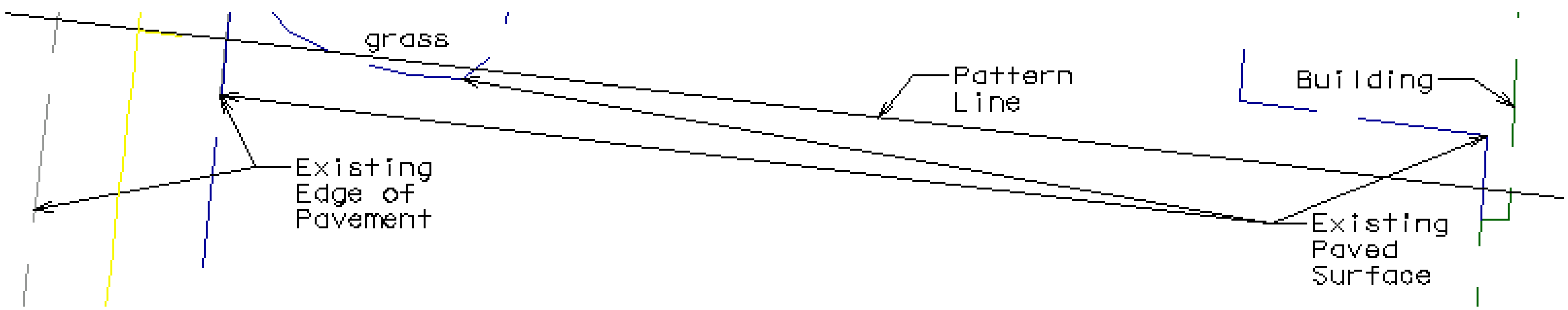
Existing Paved Surfaces

Drawn by Existing Features Typical Section.

For existing paved surfaces to be drawn in the cross section, Existing Paved Surface lines must be drawn in the designated Existing TOPO DGN and the redefinable variable **_s_ShowPavedSurface** must be set to ^yes^ for the current station. The symbology for the line is: Level 20; Color 113; Style dash 3; Weight 1. Both edges of the existing paved surface must cross the pattern line for it to be drawn correctly. It is recommend that any existing paved surface lines that are not to be used for showing existing paved surface in the cross section view be placed in a DGN other than the one defined as the Existing TOPO DGN.

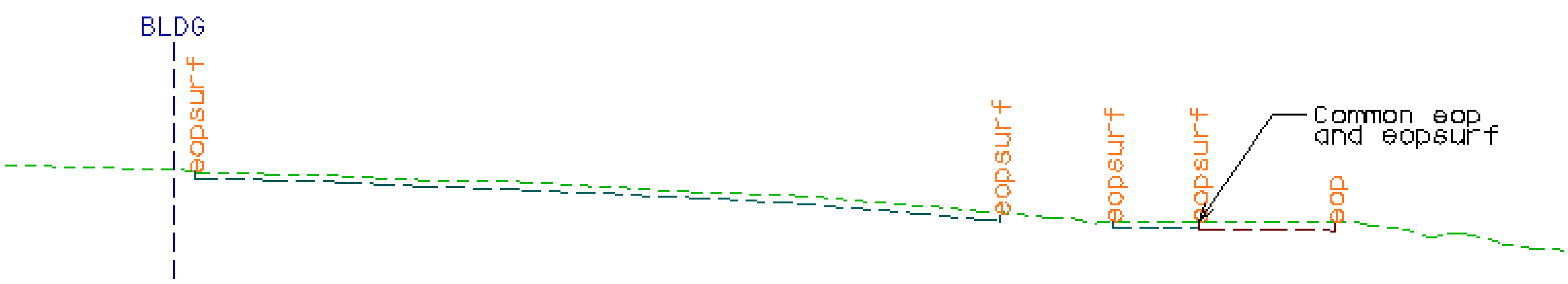
The thickness (master units) shown in the cross-section is controlled by the redefinable variable **_d_ExistPavedSurfaceThick**.

Example



Topo View

(The existing edge of pavement and the existing paved surface lines are plotted on top of each other at their common location)



Cross-Section View

Roadside

Existing Sidewalks

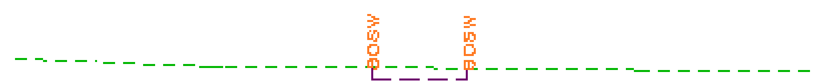
Used by Existing Features Typical Section

For an existing sidewalk to be drawn in the cross section, Existing Sidewalk lines must be drawn in the designated Existing TOPO DGN and the redefinable variable **_s_ShowSidewalk** must be set to ^yes^ for the current station. The symbology for the line is: Level 26; Color 1; Style dash 2; Weight 1. Both edges of the sidewalk must cross the pattern line for it to be drawn correctly.

The thickness (in master units) shown in the cross-section is controlled by the redefinable variable **_d_ExistSidewalkThick**.

Example

Topo View



Cross-Section View

Drainage

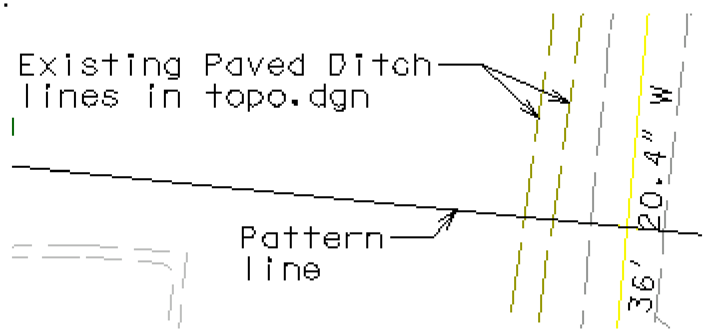
Existing Paved Ditches

Drawn by Existing Features Typical Section.

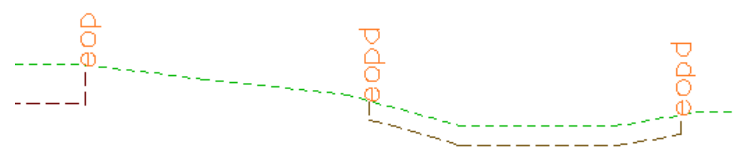
For existing paved ditches to be drawn in the cross section, Existing Paved Ditch lines must be drawn in the designated Existing TOPO DGN and the redefinable variable **_s_ShowPavedDitch** must be set to ^yes^ for the current station. The symbology for the line is: Level 41; Color 124; Style dash 4; Weight 1. Both edges of the existing paved ditch must cross the pattern line for it to be drawn correctly.

The thickness (in master units) shown in the cross-section is controlled by the redefinable variable **_d_ExistPavedDitchThick**.

Example:



Topo View



Cross-Section View

Property

Existing Buildings

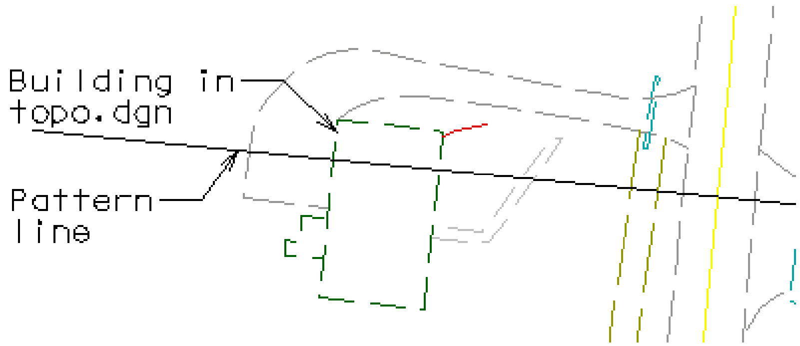
Marked by Existing Features Typical Section.

For existing building edges to be marked in the cross sections; Existing Building lines must be drawn in the designated Existing TOPO DGN and the redefinable variable **_s_ShowBuildings** must be set to ^yes^ for the current station. The symbology for the line is: Level 35; Color 14; Style dash 2; Weight 1.

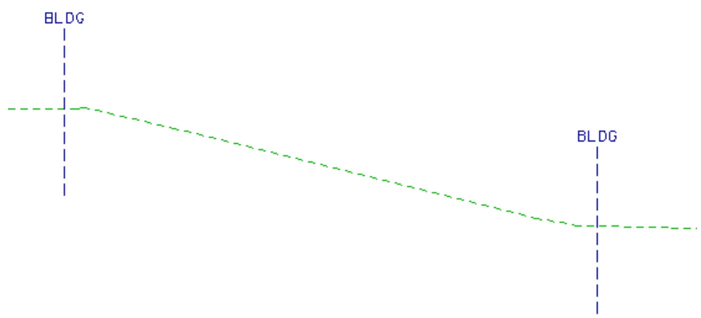
The height of the line shown in the cross-section is controlled by the redefinable variable **_d_ExistingFeatureLineHeight**, which is specified in master units.

The variable **xs scale** controls the size of the text “BLDG”.

Example



Topo View



Cross-Section View

Utilities

Existing Railroads

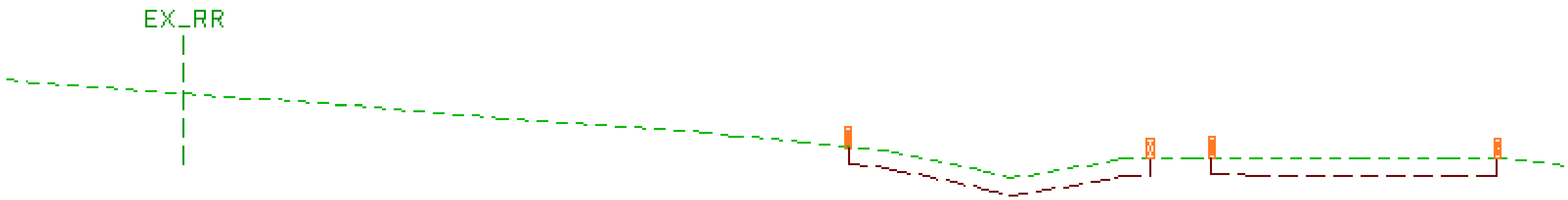
Marked by Existing Features Typical Section.

For the center line of existing railroad tracks to be marked in the cross sections; Existing Railroad lines must be drawn in the designated Existing TOPO DGN and the redefinable variable **_s_ShowRailroadTracks** must be set to ^yes^ for the current station. The symbology for the line is: Level 53; Color 1; Style RR plan single; Weight 1.

The height of the line shown in the cross-section is controlled by the redefinable variable **_d_ExistingFeatureLineHeight**, which is specified in master units.

The variable **xs scale** controls the size of the text "EX_RR".

Example



Safety and Structures

Existing Wall

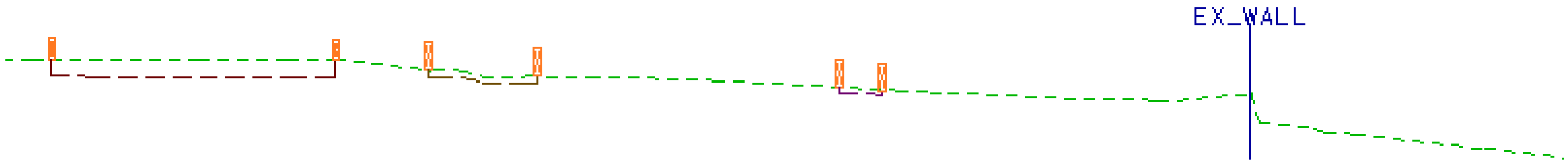
Marked by Existing Features Typical Section.

For existing retaining walls to be marked in the cross sections; Existing Wall lines must be drawn in the designated Existing TOPO DGN and the redefinable variable **_s_ShowRetainingWall** must be set to ^yes^ for the current station. The symbology for the line is: Level 23; Color 51; Style dash 3; Weight 2.

The height of the line shown in the cross-section is controlled by the redefinable variable **_d_ExistingFeatureLineHeight**, which is specified in master units.

The variable **xs scale** controls the size of the text "EX_WALL".

Example



Appendix 4 New Pavement Structure

Drawn by Undivided New Pavement Typical Section

Pavement

Edge of Pavement lines must be drawn in the designated Proposed Plan DGN for new pavement to be drawn in the cross sections. The symbology for the line is: Level 20; Color 0; Style 0; Weight 2.

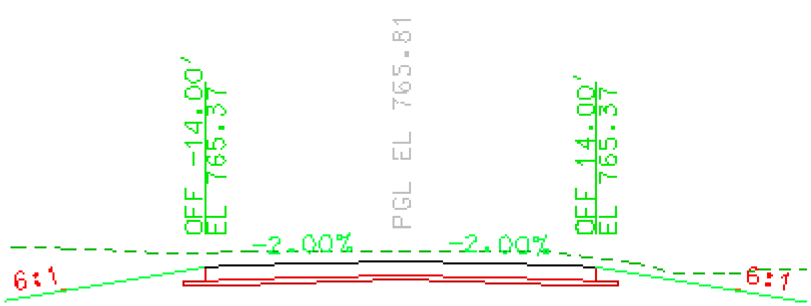
How the pavement is closed is controlled by the redefinable variable **_s_PavementType**, which must be set to either ^C^ for concrete pavement with a vertical closure or to ^B^ for bituminous pavement with a 1:1 closure.

The redefinable variable **_d_NormalPavementSlope** defines the normal pavement slope in percent on tangent sections. This is typically a -2 percent or similar. This value directly affects the way super elevation transitions are calculated. It represents normal pavement crown slope when no super elevation is present. Do not include the percent sign. Do include the negative sign when applicable. The superelevation slope is obtained from the shape cluster in the designated Shapes DGN.

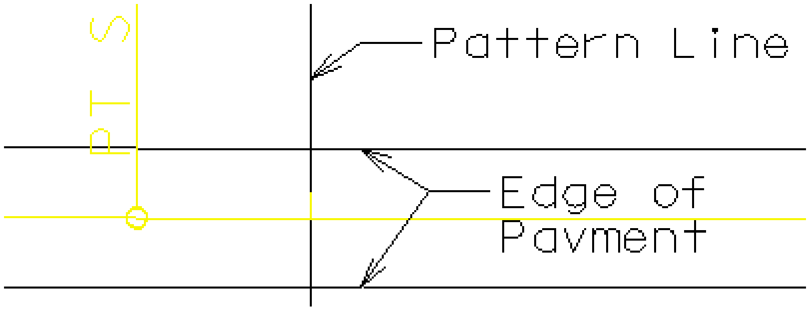
Concrete Pavement

Since concrete pavement typically has only one layer of pavement, the redefinable variable **_d_PavementLayer1Thick** is set to the full pavement depth in master units. The bottom aggregate layer will be drawn with the 18-inch extension as required by PDM section 6-03.7 paragraph (3).

Example



Cross-Section View

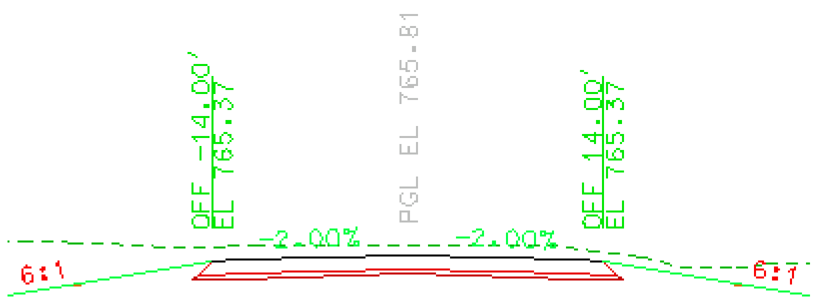


Proposed Plan View

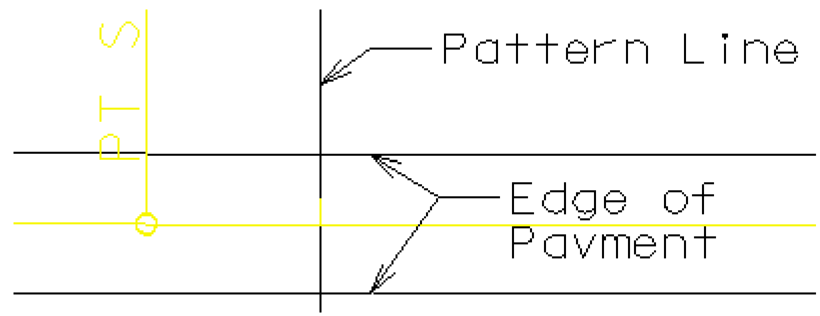
Bituminous Pavement

Bituminous pavement can be drawn as one layer by assigning the full pavement depth to **_d_PavementLayer1Thick** (in mater units) and leaving the other pavement layer thickness variables set to zero (0), as shown in the first example below. It can also be drawn with up to four (4) layers using **_d_PavementLayer1Thick**, **_d_PavementLayer2Thick**, **_d_PavementLayer3Thick**, and **_d_PavementLayer4Thick** for the top through bottom layers respectively, as shown in the bottom example. See PDM Sec 6-03.4 for more information on pavement layer thicknesses for flexible (bituminous) pavement design. Any aggregate layers have a 1:1 closure.

Example 1

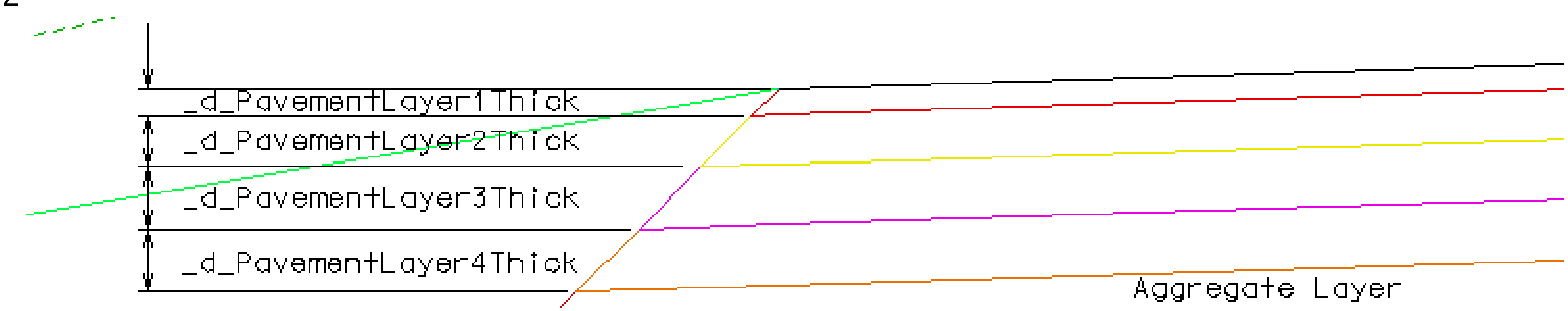


Cross-Section View



Proposed Plan View

Example 2



Cross-Section View Showing Possibility of Having Up To Four (4) Pavement Layers

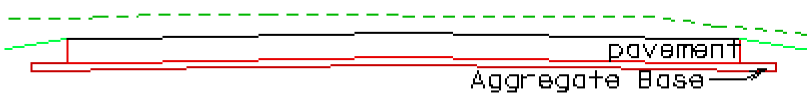
Base Considerations

The pavement base material can be drawn using either an aggregate or a rock base. See PDM Section 6-03.7 for a full discussion of which base should be used for a project. The information provided here relates only to how the criteria draws the base, not which base should be used.

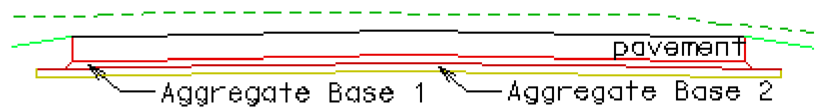
Aggregate Base

One or two aggregate base layers may be drawn. If values greater than zero assigned to both **_d_Aggbase1Thickness** and **_d_Aggbase2Thickness**, two layers will be drawn and the upper layer will be drawn as a permeable base. If only one layer is needed, **_d_Aggbase1Thickness** must be greater than zero and **_d_Aggbase2Thickness** must be set to zero. All values are in master units.

Example 1: Aggregate base under concrete (rigid) pavement



Cross-Section View of One Aggregate Base



Cross-Section View of Two Aggregate Bases

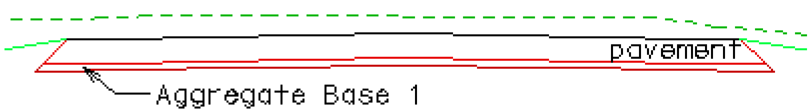
Example 2: Aggregate base extension under concrete (rigid) pavement

For concrete pavement, the 18” aggregate extension can either be drawn at the pavement or the shoulder slope, as determined by the setting for the redefinable variable **_s_ExtensionSlope**. Set it either to ^P^ for drawing the extension at the pavement slope or to ^S^ for drawing the extension at the shoulder slope. The carets "^" are required.

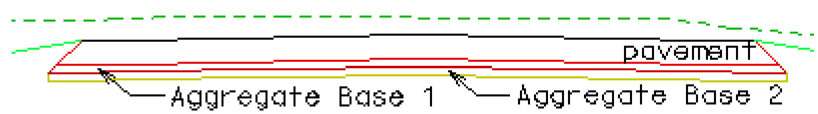
Aggregate Extension at Pavement Slope

Aggregate Extension at Shoulder Slope

Example 3: Aggregate base under bituminous (flexible) pavement



Cross-Section View of One Aggregate Base



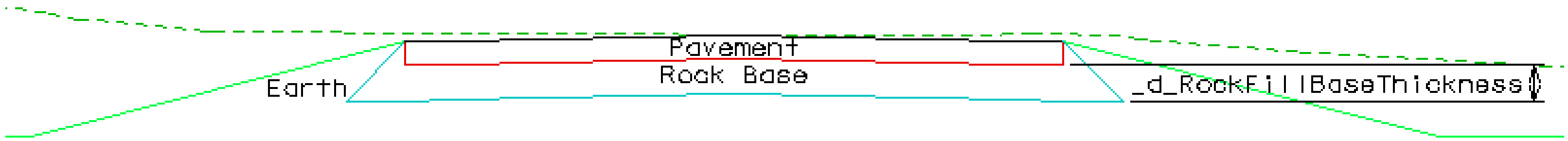
Cross-Section View of Two Aggregate Bases

Rock Base

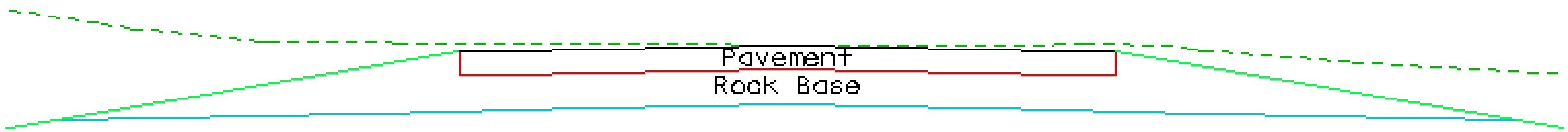
For the rock base to be drawn correctly, both **_d_Aggbase1Thickness** and **_d_Aggbase2Thickness** must be set to zero.

There are three redefinable variables that control how the rock base is drawn. The rock thickness as measured down from the bottom of the pavement in master units is set by **_d_RockFillBaseThickness**. Whether the rock base is daylighted on tangent sections and on the low side of super elevated sections is determined by **_s_RockFillBaseDaylight** (^Yes^ to daylight or ^No^ to not daylight). The last variable **_s_RockFillBaseDaylightHSS** determines if the rock base is to daylight on the high side of super elevated pavement, which according to current design practice should always be set to ^No^.

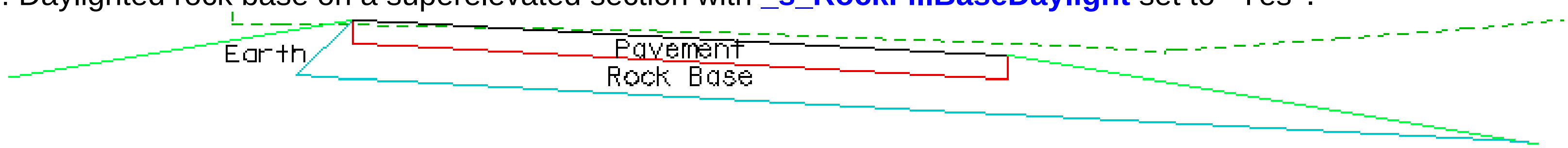
Example 1: Earth covered rock base on a superelevated section with **_s_RockFillBaseDaylight** set to ^No^.



Example 2: Daylighted rock base on a tangent section with **_s_RockFillBaseDaylight** set to ^Yes^.



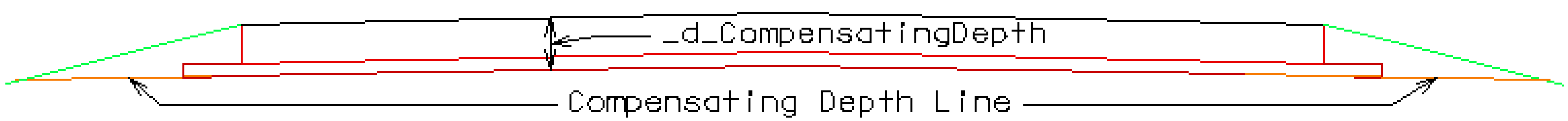
Example 3: Daylighted rock base on a superelevated section with **_s_RockFillBaseDaylight** set to ^Yes^.



Compensating Depth

A line can be drawn at a fixed distance below the pavement surface. This line is projected to proposed ground. To have this line included, set **_d_CompensatingDepth** to the desired depth in master units.

Example



Under Drains

The only underdrains drawn by the criteria are the pavement edge drains required for medium and heavy duty routes without a daylighted rock base. See PDM Section 6.02 and MoDOT Standard Plan 605.10 for further design information.

The two redefinable variables **_s_DrawLeftUnderdrain** and **_s_DrawRightUnderdrain** control whether or not edge drains are drawn on the left and right side respectively. Set the variables as ^Yes^ to draw the drain and as ^No^ to not draw them. Even if the variable is set to ^Yes^, the edge drain will **not** be drawn on the *high* side of super elevated pavement.

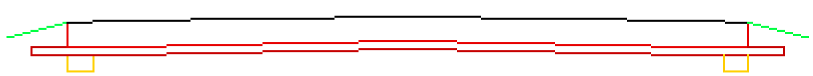
The width of the underdrain is determined by **_d_UnderdrainWidth**. Based on MoDOT Standard Plan 605.10, the width of an edge drain is always 1'; therefore leave this variable set to its default value of 1.

The depth of the drawn underdrain is controlled by the redefinable variable **_d_UnderdrainHeight** in master units and varies depending upon the number of aggregate layers, according to MoDOT Standard Plan 605.10. Each case is covered below:

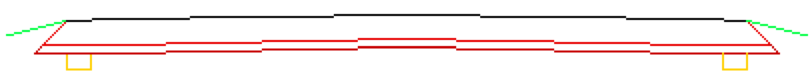
One Aggregate Layer

If there is one aggregate layer **_d_UnderdrainHeight** should be set to 8/12.

Example



Underdrain Under Concrete Pavement

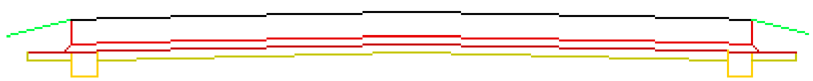


Underdrain Under Bituminous Pavement

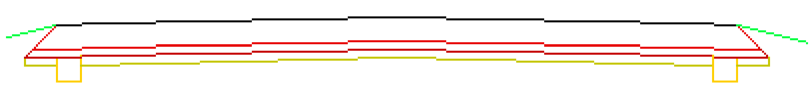
Two Aggregate Layers

If there are two aggregate layers **_d_UnderdrainHeight** should be set to 12/12.

Example



Underdrain Under Concrete Pavement



Underdrain Under Bituminous Pavement

Appendix 5 Shoulders

Drawn by Undivided New Pavement Typical Section

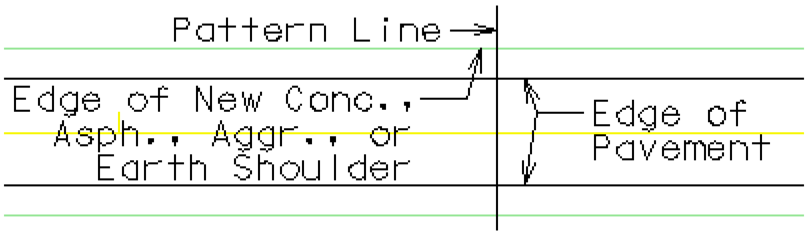
Determining Shoulder Width and Type

The width and type of shoulder can be specified in two different ways. They can be determined either an edge of shoulder line in the Proposed Plan DGN or from a redefinable variable.

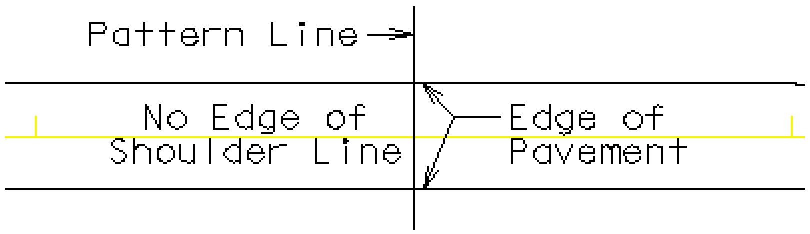
Edge of shoulder lines may be drawn in the designated Proposed Plan DGN. The symbology for the line is: Level 18; Color 11; Style 0; Weight 2. To draw the lines use the **D&C Manager** items **EOS New Conc**., **EOS New Asph**., or **EOS New Aggr**., or **EOS New Earth** depending upon the type of shoulder to be drawn. These items are located in the D&C Manager path: "**Design Standards/Roadway/**". For the line to be recognized by the criteria, it must be located within the search distance specified by the redefinable vaiables **_d_OutsideShoulderSearchDistance** and **_d_MedianShoulderSearchDistance** in master units. This distance is measured from the proposed edge of pavement. This variable is required to avoid locating a plan view shoulder element from an adjacent or parallel roadway. This number MUST be greater than zero.

If an edge of shoulder line is not found within the applicable search distance, redefinable variables are used to determine the width and type of shoulder. The variables **_d_OutsideShoulderWidth** and **_d_MedianShoulderWidth** control the width (master units) of the shoulder; while the shoulder type is controlled by **_s_OutsideShoulderType** or **_s_MedianShoulderType**, which may be set to ^C^ for concrete, ^B^ for bituminous, ^A^ for aggregate, or ^E^ for earth shoulders. A plan element will **ALWAYS** "override" these variables.

Example



Proposed Plan View with an Edge of Shoulder Line within Search Distance to Control Shoulder Width



Proposed Plan View without an Edge of Shoulder Line within Search Distance. Variable Controls Width

Shoulders In Cross Section

The user has the option of whether to label the edge of shoulder elevation and offset by setting the value for the redefinable variable **_s_LabelShoulderElevations**. Set it to either ^Yes^ or ^No^. Note that if the labels are drawn, they will be drawn as part of graphic group to allow for easy manipulation. The carets "^" are required.

Five additional redefinable variables are provided to control how the shoulders are drawn in cross section. Their values need to be set by the user based on the type of shoulder to be drawn. The following sections give the values needed for the different types of shoulder used by MoDOT, based upon PDM Figures 6-03.1 thru 6-03.9. If the PDM should be updated, please use the values from the update rather than those given below.

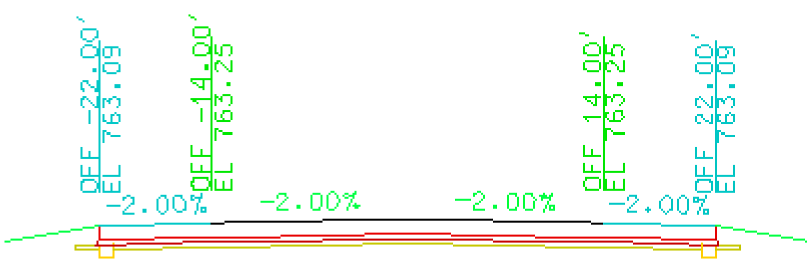
Type A and Type A1 Shoulders (Full Pavement Depth Shoulders)

Type A shoulder is used for all medium and heavy duty pavements as specified in PDM Figures 6-03.6 thru 6-03.9. Type A1 shoulder is used on all non-curbed medians (PDM Figures 6-03.4, 6-03.7, & 6-03.8). The shoulder type needs to be concrete or bituminous.

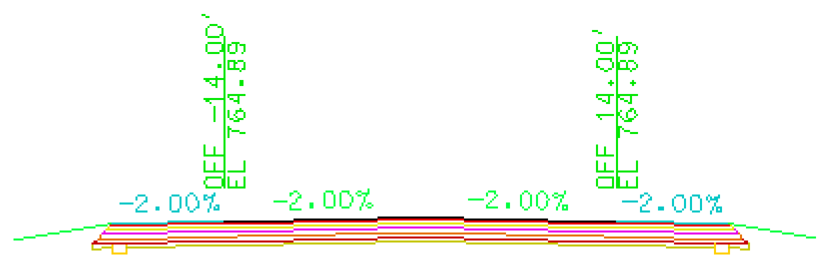
Both the outside and median shoulder slope is –2% on tangent, so the redefinable variables **_d_NormalOutsideShoulderSlope** and **_d_NormalMedianShoulderSlope** should both be set to –2.

The shoulder layer thickness values for **_d_ShoulderLayer1Thick**, **_d_ShoulderLayer2Thick**, **_d_ShoulderLayer3Thick**, and **_d_ShoulderLayer4Thick** should match the values for the corresponding pavement layer thickness. This allows for a continuation of the pavement layers between the pavement and shoulder since the cross sections do not indicate the location of any joints. It is also required to have the aggregate base under the pavement extend under the shoulder as well.

Examples



Concrete (Rigid) Pavement and Shoulder
EOS Offset and Elevation Shown



Bituminous (Flexible) Pavement and Shoulder
EOS Offset and Elevation Not Shown

Type A2 Shoulders (Partial Pavement Depth Bituminous Shoulders)

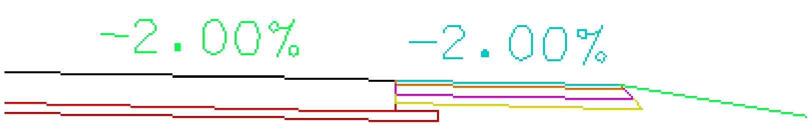
Type A2 shoulder is used on the outside of light duty pavements with ADT greater than 3500 (PDM Figures 6-03.3 thru 6-03.5). The shoulder type must be bituminous to have the correct shoulder closure.

Since all paved shoulders have a –2% cross slope, **_d_NormalOutsideShoulderSlope** should both be set to –2.

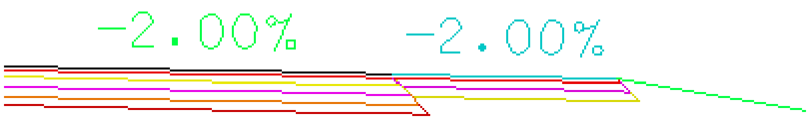
To show each of the bituminous layers, **_d_ShoulderLayer1Thick** and **_d_ShoulderLayer2Thick** should be set to 1.75/12 and 4/12, respectively.

The shoulder aggregate layer obtains its thickness from **_d_ShoulderLayer3Thick**. If the pavement has a rock base, it should be set to the total pavement thickness in master units minus 5.75/12 so that the shoulder aggregate extends down to the rock base. When a rock base is not present, **_d_ShoulderLayer3Thick** is 4/12. For this shoulder type **_d_ShoulderLayer4Thick** will always be zero (0).

Example 1 Pavement and Shoulder Without Rock Base

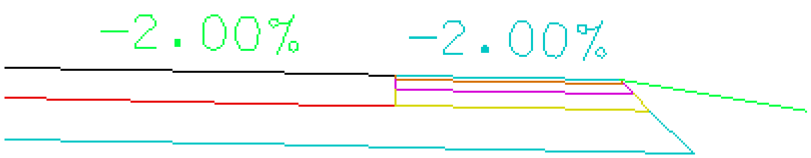


Concrete (Rigid) Pavement and Type A2 Shoulder

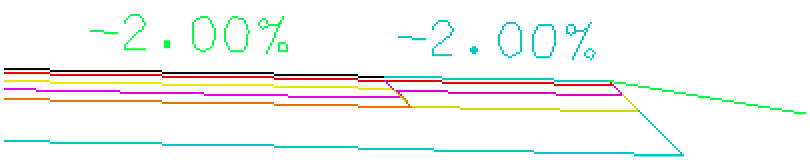


Bituminous (Flexible) Pavement and Type A2 Shoulder

Example 2 Pavement and Shoulder on Rock Base



Concrete (Rigid) Pavement and Type A2 Shoulder



Bituminous (Flexible) Pavement and Type A2 Shoulder

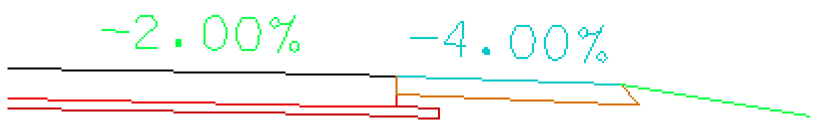
Type B Shoulders (8” Aggregate Shoulders)

Type B shoulder is for light duty pavement & ADT of 1700-3500 (PDM Figure 6-03.2). The shoulder type must be aggregate and **_s_ExtensionSlope** should be set to ^S^ shoulder slope.

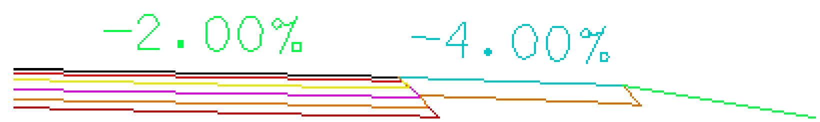
Since all non-paved shoulders have a –4% cross slope, **_d_NormalOutsideShoulderSlope** should both be set to –4.

Since the aggregate is 8” thick, **_d_ShoulderLayer1Thick** should be 8/12. The redefinable variable values for the rest of the layers are ignored (**_d_ShoulderLayer2Thick**, **_d_ShoulderLayer3Thick**, & **_d_ShoulderLayer4Thick**).

Example



Concrete (Rigid) Pavement and Type B Shoulder



Bituminous (Flexible) Pavement and Type B Shoulder

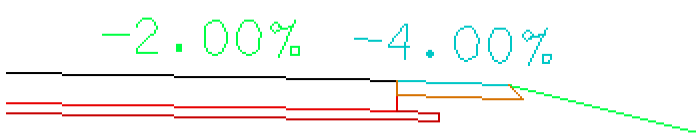
Type C Shoulders (6” Aggregate Shoulders)

Type C shoulder is for light duty pavement & ADT of 750-1700 (PDM Figure 6-03.1). The shoulder type must be aggregate and **_s_ExtensionSlope** should be set to ^S^ shoulder slope.

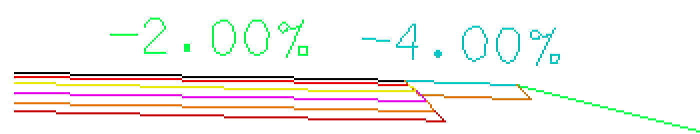
Since all non-paved shoulders have a –4% cross slope, **_d_NormalOutsideShoulderSlope** should both be set to –4.

Since the aggregate is 6” thick, **_d_ShoulderLayer1Thick** should be 6/12. The redefinable variable values for the rest of the layers are ignored (**_d_ShoulderLayer2Thick**, **_d_ShoulderLayer3Thick**, & **_d_ShoulderLayer4Thick**).

Example



Concrete (Rigid) Pavement and Type C Shoulder



Bituminous (Flexible) Pavement and Type C Shoulder

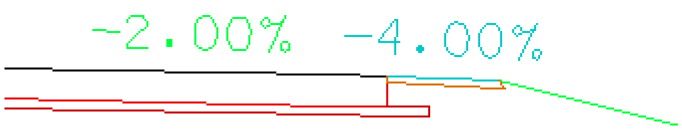
Type D Shoulders (3" Minimum Aggregate Shoulders)

Type D shoulder is for light duty pavement & ADT of 400-750 (PDM Figure 6-03.1). The shoulder type must be aggregate and **_s_ExtensionSlope** should be set to ^P^ Pavement slope.

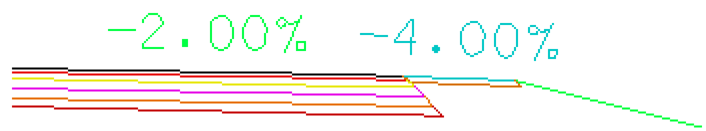
Since all non-paved shoulders have a –4% cross slope, **_d_NormalOutsideShoulderSlope** should both be set to –4.

Since the minimum aggregate thickness is 3", **_d_ShoulderLayer1Thick** should be set to a minimum of 3/12. The redefinable variable values for the rest of the layers are ignored (**_d_ShoulderLayer2Thick**, **_d_ShoulderLayer3Thick**, & **_d_ShoulderLayer4Thick**).

Example



Concrete (Rigid) Pavement and Type D Shoulder



Bituminous (Flexible) Pavement and Type D Shoulder

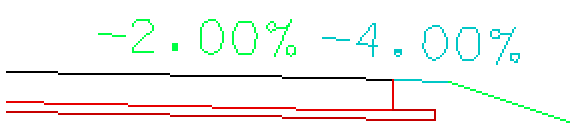
Type E Shoulders (Earth Shoulders)

Type E shoulder is for light duty pavement & ADT < 400 (PDM Figure 6-03.1) and **_s_ExtensionSlope** should be set to ^P^ Pavement slope.

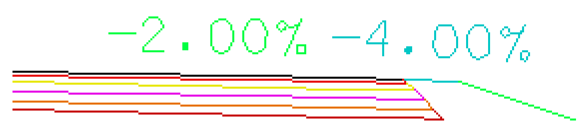
Since all non-paved shoulders have a –4% cross slope, **_d_NormalOutsideShoulderSlope** should both be set to –4.

The shoulder type should be set to earth so that the layer redefinable variables **_d_ShoulderLayer1Thick**, **_d_ShoulderLayer2Thick**, **_d_ShoulderLayer3Thick**, & **_d_ShoulderLayer4Thick** are ignored.

Example



Concrete (Rigid) Pavement and Type E Shoulder



Bituminous (Flexible) Pavement and Type E Shoulder

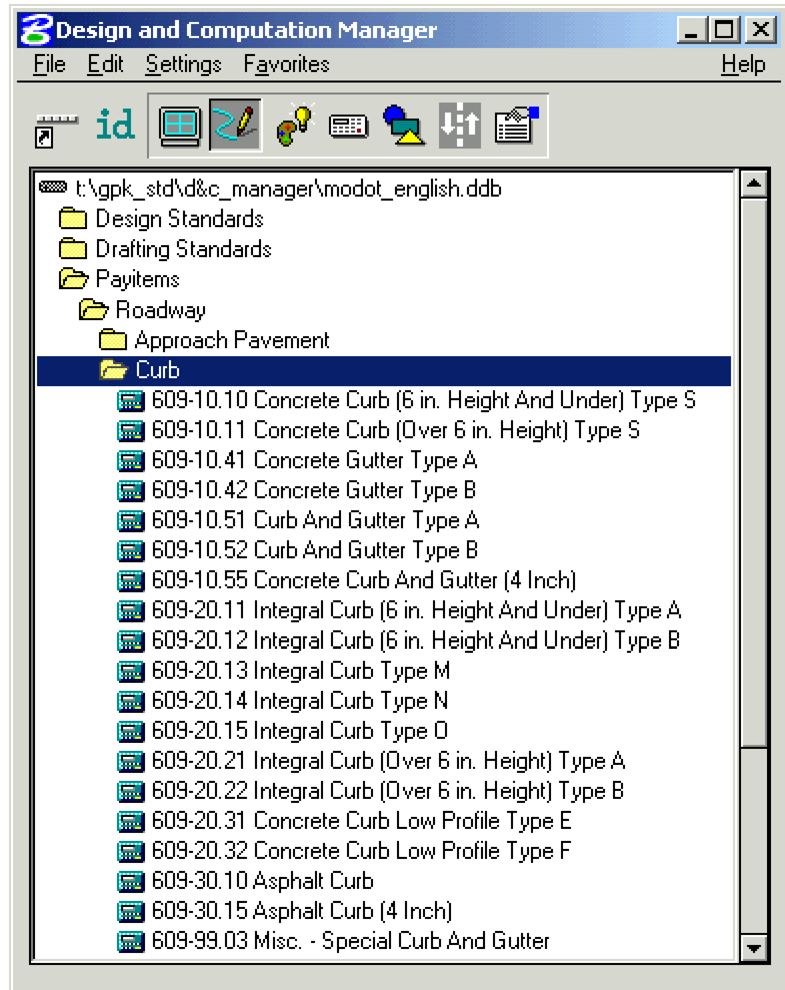
Appendix 6 Curbing and Urban Shoulders (U1 & U2)

Drawn by Undivided New Pavement Typical Section

Plan View Geometry

Curbing

The type of curb drawn in a proposed cross section is determined by an element in the defined Proposed Plan DGN. This line must have the attribute matching the pay item for the desired type of curb, which is obtained from the **Design and Computation Manager** item in the **Payitems/Roadway/Curb** folder shown in the figure to the right. The line must be located within a specified curb search distance from the proposed edge of paved surface. For curb and gutter (Types A & B) this may be the edge of pavement or the edge of shoulder. For all other curb types, the edge of paved surface must be a shoulder.



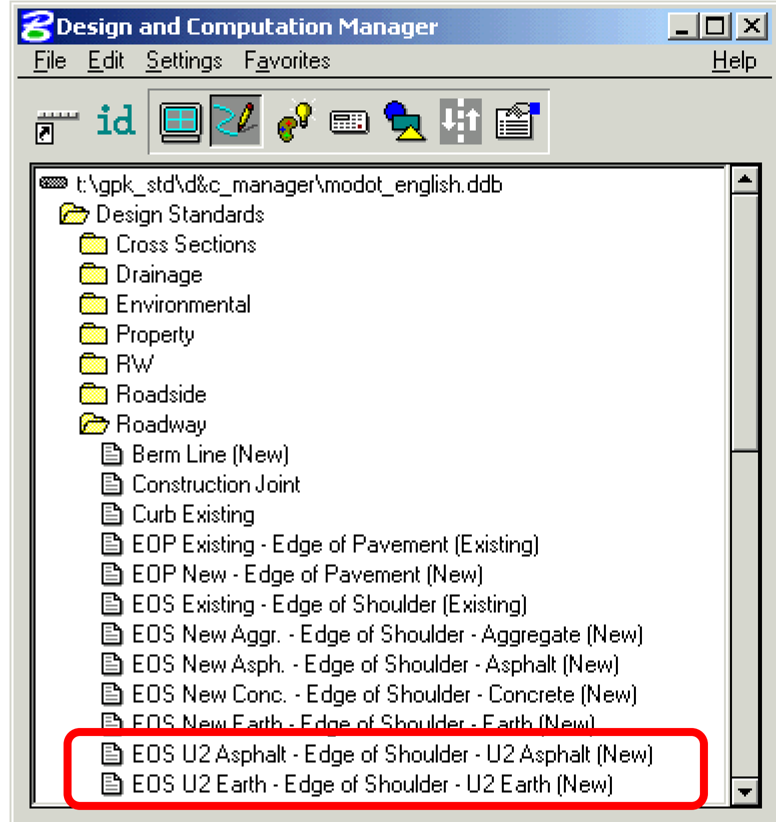
The specified curb search distance is obtained from the redefinable variable: **_d_CurbSearchDistance**, which must be a positive number in master units.

The location of the curb in the proposed cross section is determined by the proposed edge of paved surface in the Proposed Plan DGN. This line indicates the start of the gutter section for a curb & gutter (Type A, B, or Misc. - Special) or the gutter line for an integral (Type A, B, F, M, N, or O) or a separated curb (Type F or S). Consequently, the curb line shown in the Proposed Plan DGN can be drawn at either the front or the back of the curb without affecting the location of the curb in the proposed cross section.

U1 Shoulders

U1 shoulders use the same parameters as **Type A, A1, or A2 shoulders**. As with rural shoulders, the width and type is obtained either from a plan view element or from the redefinable variables. See Appendix 5 for further details.

U2 Shoulders

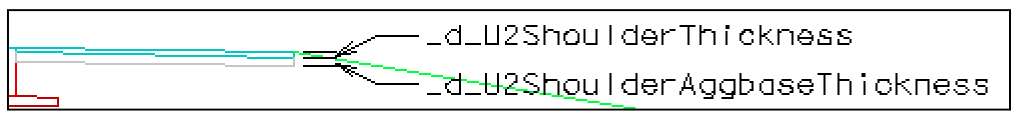


The presence of a U2 shoulder behind the curb can be indicated in two ways: either by placing an edge of U2 shoulder line in the Proposed Plan DGN or by defining a U2 shoulder width using a redefinable variable.

To draw U2 shoulder lines in the designated Proposed Plan DGN, use the **Design and Computation Manager** items EOS U2 Asphalt or EOS U2. These items (outlined in the figure to there right) are in the **Design Standards/Roadway** folder. For the line to be recognized by the criteria, it must be located within the search distance (master units) specified by the redefinable variable **_d_U2ShoulderSearchDistance**, as measured from the back of the curb. This variable is required to avoid locating a plan view U2 shoulder element from an adjacent or parallel roadway. This number MUST be greater than zero.

If a U2 edge of shoulder line is not found within the applicable search distance, the criteria will use **_d_U2ShoulderWidth** to determine the width of the U2 shoulder in master units from the back of the curb. If a U2 shoulder is not needed, do not have a U2 shoulder line within the search distance in the Proposed Plan DGN and set _d_U2ShoulderWidth to zero. The plan element will **ALWAYS** "override" this variable.

If an EOS U2 Asphalt line or _d_U2ShoulderWidth is used to specify the width of the U2 shoulder, subsurface shoulder lines may be drawn in the proposed cross section by specifying the appropriate thicknesses (in master units) for the Redefinable Variables **_d_U2ShoulderThickness** (for the asphalt thickness) and **_d_U2ShoulderAggbaseThickness** (for the aggregate base thickness). If an EOS U2 Earth line is used to indicate the presence of a U2 shoulder, the values of the two thickness variables are ignored and only the surface line is drawn.



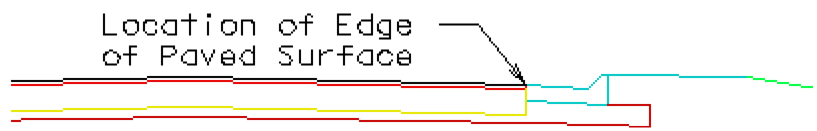
The only type of curb that can be drawn behind the U2 shoulder is Type S (Separated) curb. If this type of curb is desired, include a line in the Proposed Plan DGN using the appropriate D&C curb pay item on the outside of the U2 shoulder within the search distance defined by **_d_CurbSearchDistance**, which must be less than the width of the U2 shoulder and is measured from the outer edge of the U2 shoulder. The face of the curb will be drawn at the outer edge of the U2 shoulder in the proposed cross section.

Curbing Details

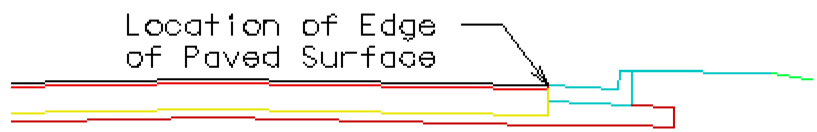
Curb & Gutter

Curb & gutter can be placed adjacent to any paved surface (pavement or shoulder), which can be either concrete or asphalt. If the curb & gutter is placed next to a shoulder, the shoulder thicknesses must match the pavement thickness for the pavement base aggregate to extend under the curb & gutter. The outer edge of paved surface locates the inner face of the gutter in the proposed cross section.

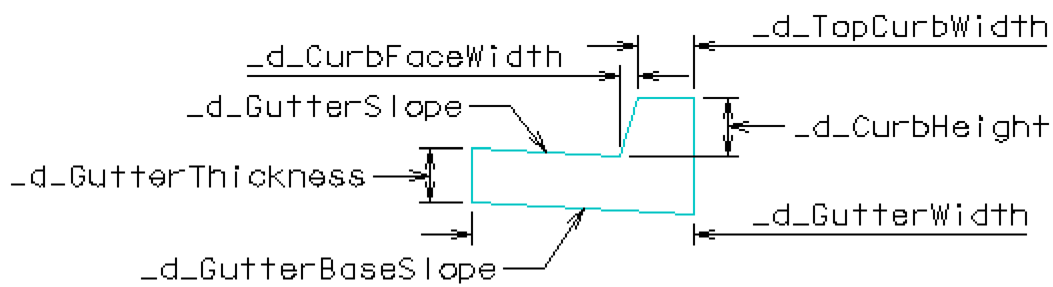
Type A (Mountable) Curb & Gutter
In the Proposed Plan DGN, the line for Type A Curb and Gutter needs to have the attribute of pay item 609-10.51.



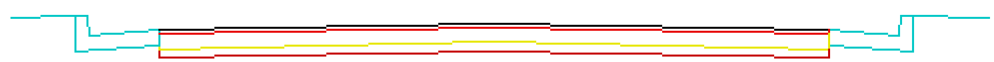
Type B (Barrier) Curb & Gutter
In the Proposed Plan DGN, the line for Type B Curb and Gutter needs to have the attribute of pay item 609-10.52.



User Defined Curb & Gutter
The user can define a special curb and gutter by placing a line in the Proposed Plan DGN having the attribute of pay item 609-99.03. If this line is within the search distance the criteria will draw a non-standard curb and gutter using the redefinable variables shown in the figure to the right. The values for _d_GutterSlope and _d_GutterBaseSlope are in percent (%) and all other values are in master units.



Warning for Curb & Gutter
If the bottom of the pavement is not below the bottom of the curb and gutter, the criteria will not extend the aggregate under the curb and cutter. Instead, the aggregate layers will stop at the face of the gutter and the following note will be plotted in the cross-section: Warning, AggBase1 Depth Not Able to Extend Under Curb! It is assumed that the pavement structure will be at least as thick and the gutter.



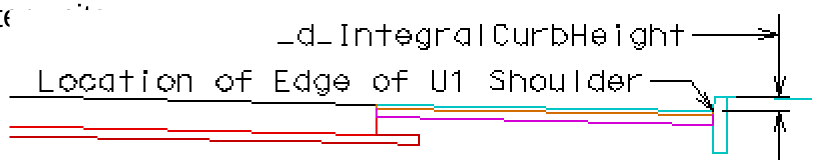
Separated Curb

In the Proposed Plan DGN, the curb line with the appropriate pay item attribute can be anywhere within the curb search distance from the edge of a shoulder, which indicates where the gutter line will be drawn in the Proposed Cross-Section DGN.

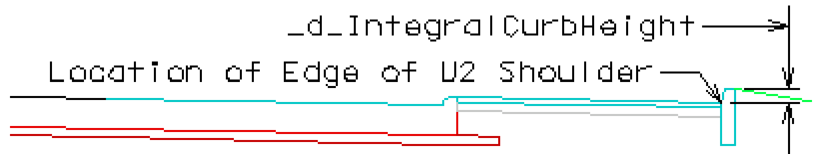
Note: Even though the Standards Plans allow for the height of the curb below the paved surface to be less than the default value if the curb is keyed into rock, the default value is always used.

Type S Separated Barrier Curb

In the Proposed Plan DGN, the line for the Type S Curb needs to have the attribute of pay item 609-10.10 or 609-10.11, which can be placed after either a U1 **or** a U2 shoulder. If the criteria file finds a curb line having either pay item within the search distance, it will draw a Type S (Separated) Curb using the value of the Redefinable Variable **_d_IntegralCurbHeight** to specify the curb height in maste



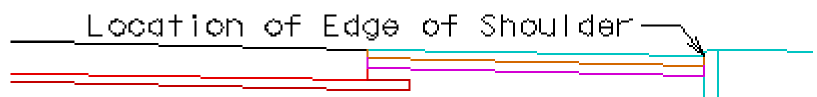
Type S Curb After a U1 Shoulder



Type S Curb After a U2 Shoulder

Type F (Separated) Low Profile Curb

In the Proposed Plan DGN, the line for the Type F Curb needs to have the attribute of pay item 609-20.32.



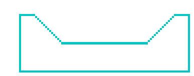
Type F (Separated) Low Profile Curb

Gutter

Gutter may be drawn after a shoulder or a retaining wall in cut. The outer edge of the shoulder or the back of the retaining wall locates the start of the gutter. If a gutter line with the appropriate pay item attribute is found anywhere within the curb search distance in the Proposed Plan DGN, gutter will be drawn in the Proposed Cross-Section DGN.

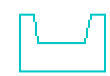
Type A Gutter

In the Proposed Plan DGN, the line for the Type A Gutter needs to have the attribute of pay item 609-10.41.


Type A Gutter

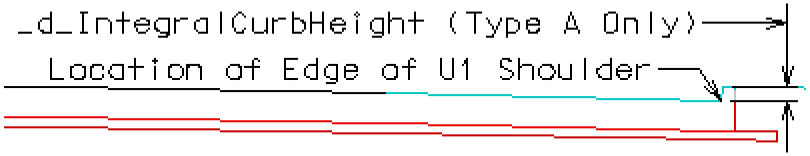
Type B Gutter

In the Proposed Plan DGN, the line for the Type B Gutter needs to have the attribute of pay item 609-10.42.

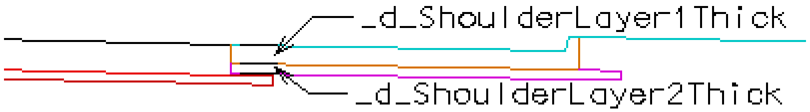

Type B Gutter

Integral Curb Types A, M, O, and E

In the Proposed Plan DGN, the curb line with the appropriate pay item attribute can be anywhere within the curb search distance from the edge of shoulder, which indicates where the gutter line will be drawn in the Proposed Cross-Section DGN. To be drawn correctly, all types of integral curb require a concrete U1 shoulder to be defined. The thickness of the U1 shoulder is controlled by the redefinable variable **_d_ShoulderLayer1Thick**. If its value matches **_d_PavementLayer1Thick**, the aggregate under the pavement will extend under both the shoulder and the curb. If the thickness of the U1 shoulder is different from the pavement, aggregate base(s) can be drawn under the shoulder and curb by specifying a value greater than zero (0) for the thickness of the appropriate shoulder layer(s) using the redefinable variables **_d_ShoulderLayer2Thick**, **_d_ShoulderLayer3Thick**, and **_d_ShoulderLayer4Thick**.



U1 Shoulder Thickness Equals Pavement Thickness



U1 Shoulder Thickness Less Than Pavement Thickness

Type A Integral Barrier Curb

In the Proposed Plan DGN, the line for the Type A Integral Curb needs to have the attribute of pay item 609-20.11 or 609-20.21. If the criteria file finds a curb line having either pay item within the search distance, it will draw a Type A Integral Curb using the value of the Redefinable Variable **_d_IntegralCurbHeight** to specify the curb height in master units.

Type E (Integral) Low Profile Curb

In the Proposed Plan DGN, the line for the Type A Integral Curb needs to have the attribute of pay item 609-20.31

Type M (Integral) Mountable Curb

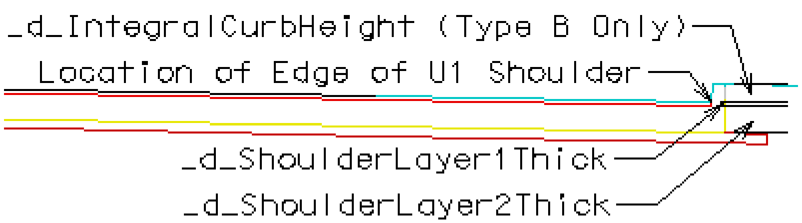
In the Proposed Plan DGN, the line for the Type M Integral Curb needs to have the attribute of pay item 609-20.13.

Type O (Integral) Mountable Curb

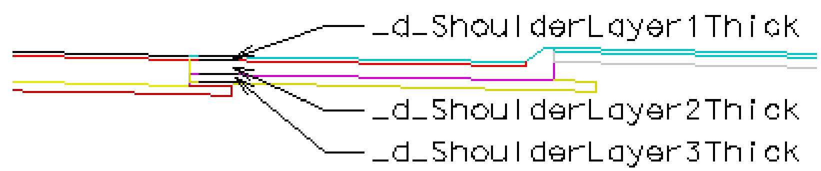
In the Proposed Plan DGN, the line for the Type O Integral Curb needs to have the attribute of pay item 609-20.15.

Integral Curb Types B and N

In the Proposed Plan DGN, the curb line with the appropriate pay item attribute can be anywhere within the curb search distance from the edge of shoulder, which indicates where the gutter line will be drawn in the Proposed Cross-Section DGN. To be drawn correctly, all types of integral curb require a concrete U1 shoulder to be defined. The total thickness of the U1 shoulder is the algebraic sum of the redefinable variables **_d_ShoulderLayer1Thick** and **_d_ShoulderLayer2Thick**, where the first variable controls the thickness of the overlay and the second controls the thickness of the concrete portion of the shoulder. For the aggregate base under the pavement to extend under the U1 shoulder and the curb, the thicknesses of the pavement and corresponding shoulder layers must be the same (**_d_PavementLayer1Thick** equals **_d_ShoulderLayer1Thick** and **_d_PavementLayer2Thick** equals **_d_ShoulderLayer2Thick** for that station). If the respective pavement and shoulder layers have different thicknesses, aggregate base(s) can be drawn under the shoulder and curb by specifying a value greater than zero (0) for the thickness of the appropriate shoulder layer(s) using the redefinable variables **_d_ShoulderLayer3Thick** and **_d_ShoulderLayer4Thick**.



U1 Shoulder Thicknesses Equal
Pavement Thicknesses



U1 Shoulder Thicknesses Do Not
Equal Pavement Thicknesses

Type B Integral Barrier Curb

In the Proposed Plan DGN, the line for the Type B Integral Curb needs to have the attribute of pay item 609-20.12 or 609-20.22. If the criteria file finds a curb line having either pay item within the search distance, it will draw a Type B Integral Curb using the value of the Redefinable Variable **_d_IntegralCurbHeight** to specify the curb height in master units.

Type N (Integral) Mountable Curb

In the Proposed Plan DGN, the line for the Type N Integral Curb needs to have the attribute of pay item 609-20.14.

Appendix 7 Standard Side Slopes (No Special Profiles)

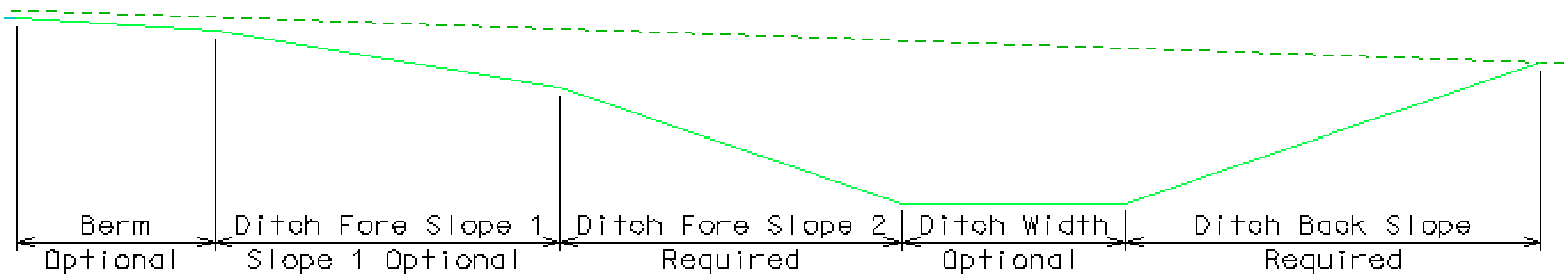
Drawn by Undivided New Pavement Typical Section.

Standard Side Slope Closure Options

The side slope can close to existing ground using a standard section in cut and fill or a special forced closure section. These options are briefly described on the rest of this page with further details provided in the rest of this appendix. The typical section may also include other items, such as a sidewalk, retaining wall, interception ditch, and levee. These options are described in other appendices.

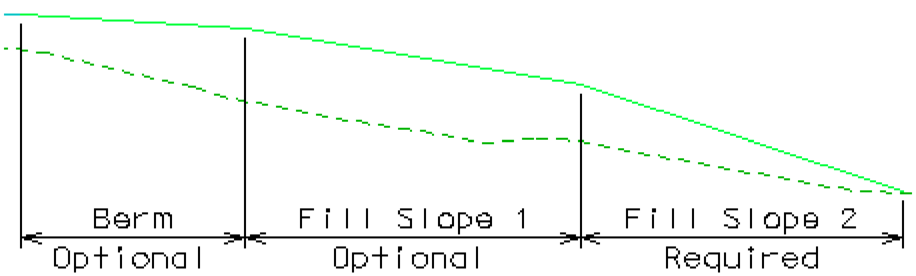
Standard Cut Slope Options

A ditch fore slope and back slope are required if a standard cut section is used, which are labeled as Ditch Fore Slope 2 and Ditch Back Slope in the following figure. In addition the user can include a berm, an additional ditch fore slope, and a ditch width.



Standard Fill Slope Options

A single fore slope is required if a standard fill section is used, which is labeled as Fill Slope 1 in the figure to the right. In addition the user can include a berm and an additional fore slope.



Standard Forced Closure Slope Option

The user can force the side slope to close directly to existing ground from the edge of shoulder or back of curb by using this option.

Side Slope Details

Berm (Optional)

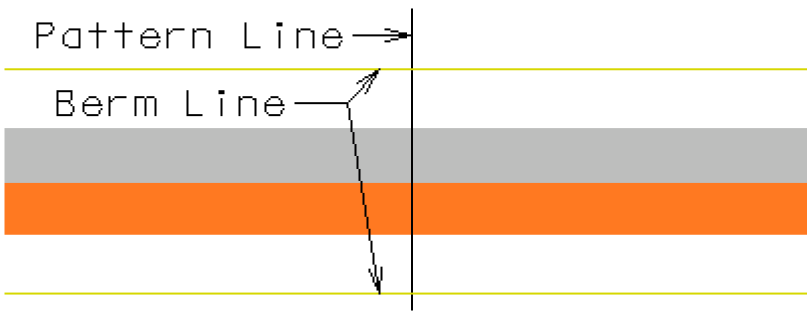
The berm option is intended to be used for non-paved guardrail widening, grass areas for locating sidewalk, and other cases that require a slope defined by percent at the start of the side slope. It may be used in cut or fill.

Berm Width

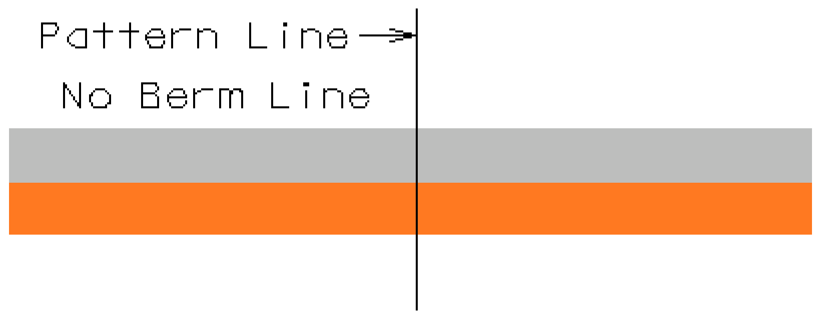
The width of the berm can be specified in two ways. It can be determined either by a berm line in the GEOPAK Lines DGN or by a redefinable variable.

Berm lines may be drawn in the designated GEOPAK Lines DGN. To draw the lines use the **D&C Manager** item **Berm Line**, which is located in the D&C Manager path: "**Design Standards/Roadway/**".

If a berm line is not present, the redefinable variables **_d_BermWidth_Left** and **_d_BermWidth_Right** are used to determine the width of the berm in master units on the left and right side, respectively. A plan element will **ALWAYS** "override" these variables.



GEOPAK Lines DGN with a Berm Line
DGN Element Controls Berm Width



GEOPAK Lines DGN without a Berm Line
Redefinable Variable Controls Berm Width

If a berm is not desired at a particular location, do not have a berm line crossing the pattern line on that side of the roadway and set the berm width redefinable variable for that location to zero.

Berm Slope

The berm slope on each side of the roadway is controlled by the redefinable variables **_d_BermSlope_Left** and **_d_BermSlope_Right** (in percent without the percent sign). Include the negative sign when applicable.

Ditch Fore Slope 1 (Optional)

The ditch fore slope 1 option is provided if a change is the run:rise defined slope is required before the ditch.

The width of the ditch fore slope 1 can be specified in two ways. It can be determined either by a cut slope 1 break line in the GEOPAK Lines DGN or by a redefinable variable.

Cut slope 1 break lines may be drawn in the designated GEOPAK Lines DGN. To draw the lines use the **D&C Manager** item **Slope Cut**, which is located in the D&C Manager path: "**Design Standards/Drainage/**".

If a slope cut line is not present, the redefinable variables **_d_DitchForeSlope1Width_Left** and **_d_DitchForeSlope1Width_Right** are used to determine the width of the ditch fore slope 1 in master units on the left and right side, respectively. A plan element will **ALWAYS** "override" these variables.

If a ditch fore slope 1 is not desired at a particular location, do not have a slope cut line crossing the pattern, set the ditch fore slope 1 width redefinable variable to zero, or make ditch fore slope 1 the same as ditch fore slope 2.

The ratio of ditch fore slope 1 is controlled by the redefinable variables **_d_DitchForeSlope1_Left** and **_d_DitchForeSlope1_Right** in Run:Rise format. Rise is ALWAYS a negative number. Example, 4:-1. Make sure to include the colon.

Ditch Fore Slope 2 (Required)

The ratio of ditch fore slope 2 is controlled by the redefinable variables **_d_DitchForeSlope2_Left** and **_d_DitchForeSlope2_Right** in Run:Rise format. Rise is ALWAYS a negative number. Example, 4:-1. Make sure to include the colon. This slope will be continued from its start to the bottom of the ditch.

Standard Ditch Depth

The depth a standard ditch is controlled by **_d_StandardDitchDepth_Left** and **_d_StandardDitchDepth_Right**, which is measured from the start of the ditch fore slope in master units. This is a positive value and MUST be greater than zero.

Ditch Width

The width of the ditch is controlled by the redefinable variables **_d_DitchWidth_Left** and **_d_DitchWidth_Right** in master units. Set the value of the variable to zero for a "V" ditch or greater than zero for a flat bottom ditch.

Ditch Back Slope (Required)

The ratio of ditch back slope is controlled by the redefinable variables **_d_DitchBackSlope_Left** and **_d_DitchBackSlope_Right** in Run:Rise format. Rise is ALWAYS a positive number. Example, 3:1. Make sure to include the colon. This slope will be continued from bottom of the ditch to existing ground for standard ditches unless Right of Way constraints are used, which are described on the next page.

Fill Slope 1 (Optional)

The fill slope 1 option is provided if a change is the run:rise defined slope is required in fill.

The width of fill slope 1 can be specified in two ways. It can be determined either by a fill slope 1 break line in the GEOPAK Lines DGN or by a redefinable variable.

Fill slope 1 break lines may be drawn in the designated GEOPAK Lines DGN. To draw the lines use the **D&C Manager** item **Slope Fill**, which is located in the D&C Manager path: "**Design Standards/Drainage/**".

If a slope fill line is not present, the redefinable variables **_d_FillSlope1Width_Left** and **_d_FillSlope1Width_Right** are used to determine the width of the fill slope 1 in master units on the left and right side, respectively. A plan element will **ALWAYS** "override" these variables.

If fill slope 1 is not desired at a particular location, do not have a slope fill line crossing the pattern, set the fill slope 1 width redefinable variable to zero, or make the slope fill slope 1 the same as fill slope 2.

The ratio of fill slope 1 is controlled by the redefinable variables **_d_FillSlope1_Left** and **_d_FillSlope1_Right** in Run:Rise format. Rise is ALWAYS a negative number. Example, 4:-1. Make sure to include the colon.

Fill Slope 2 (Required)

The ratio of fill slope 2 is controlled by the redefinable variables **_d_FillSlope2_Left** and **_d_FillSlope2_Right** in Run:Rise format. Rise is ALWAYS a negative number. Example, 4:-1. Make sure to include the colon. This slope will be continued from its start to the intersection with existing ground unless Right of Way constraints are used.

Draw Ditches In Plan View

The ditches and the flow arrows can be drawn in the plan view with processing the proposed cross-sections. The redefinable variable **_s_DrawDitchesinPlanView** controls this behavior. Set it to ^Yes^ or ^No^. The carets "^" are required.

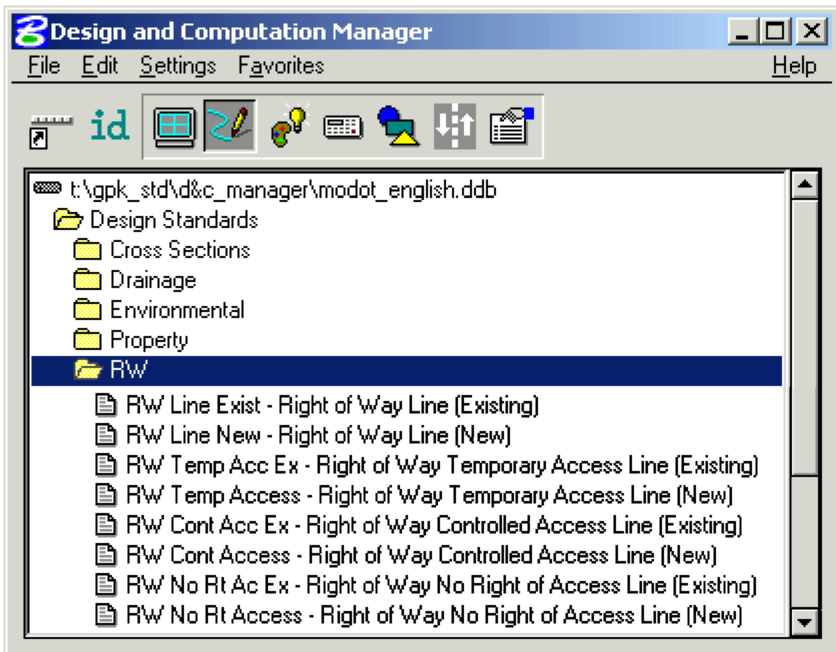
Right of Way Constrained Tie Slopes

Tie slopes can be forced to stay within the right of way. This behavior is controlled by both plan view elements drawn in the defined **Right of Way DGN** and redefinable variables.

Right of Way DGN Geometry

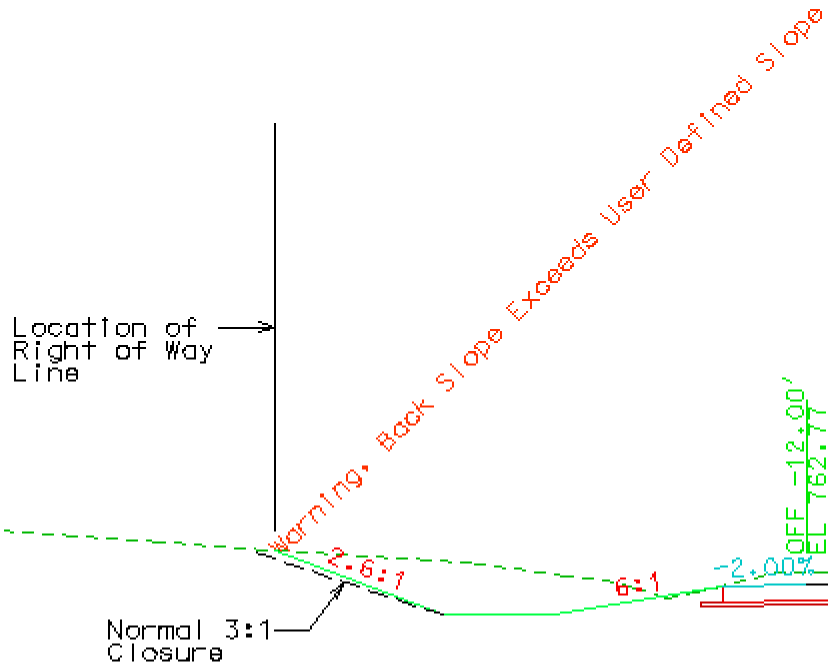
For right of way constrained tie slopes to work, a right of way line must be drawn in the defined **Right of Way DGN** using one of the eight items in the D&C Manager path: "**Design Standards/RW/**" shown below:

RW Line Exist. – Right of Way Line (Existing)
RW Line New – Right of Way Line (New)
RW Temp Acc Ex – Right of Way Access Line (Existing)
RW Temp Access – Right of Way Access Line (New)
RW Cont Acc Ex – Right of Way Controlled Access Line (Existing)
RW Cont Access – Right of Way Controlled Access Line (New)
RW No Rt Ac Ex – Right of Way No Right of Access Line (Existing)
RW No Rt Access – Right of Way No Right of Access Line (New)

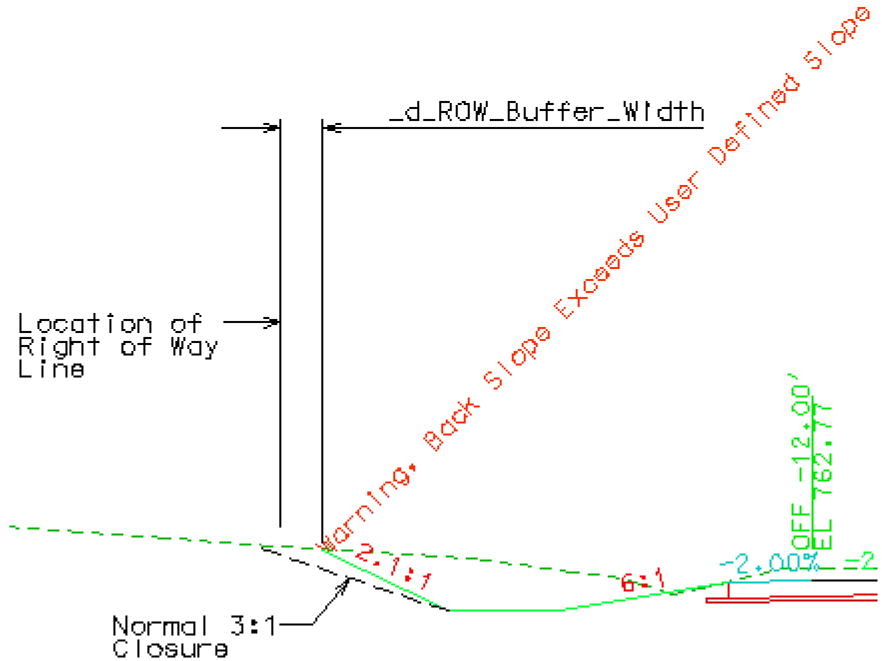


Redefinable Variables

Three redefinable variables are also used to determine how tie slopes are constrained by the right of way lines discussed above. The variables, **_d_LefROW_ConstrainedSlope** and **_d_LefROW_ConstrainedSlope** control whether or not the constraint is to be applied to the left and right sides respectively. To constrain the slope set the variable for that side to ^Yes^. If normal closure is desired, set the variable to ^No^. The carets "^" are required. An additional buffer can be applied that will force the tie to be within the right of way point by the buffer width set using the variable **_d_ROW_Buffer_Width** in master units. If the constraint is applied and the normal slope can fit within the right of way then the normal slope will be used. If the normal slope will not fit within right of way, then a steeper slope will be drawn and labeled as such. This will allow the user to identify when their slopes drawn were steeper than the defaults, as shown in the following two figures.



Right of Way Constrained Slope
With Buffer Width Set to Zero



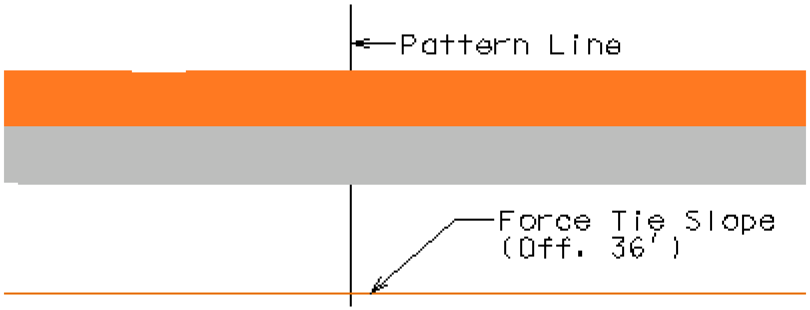
Right of Way Constrained Slope
With Buffer Width Set to 3'

Forced Closure Slopes

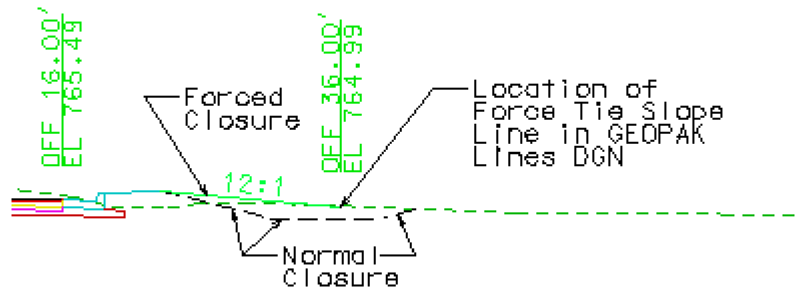
Whether or not the side slope closes directly to existing ground from the edge of shoulder or back of curb can be controlled in two ways. It can be controlled either by a **Force Tie Slope** line in the GEOPAK Lines DGN or by a redefinable variable.

Forced Closure using GEOPAK Lines DGN Element

Force Tie Slope lines may be drawn in the designated GEOPAK Lines DGN. To draw the lines use the **D&C Manager** item **Force Tie Slope**, which is located in the D&C Manager path: "**Design Standards/Roadway/**". If this line crosses the pattern line for a specific side slope, a single side slope line will be drawn from the edge of shoulder or back of curb to the location where the force tie slope line intersects the existing ground.



GEOPAK Lines DGN with Force Tie Slope Line

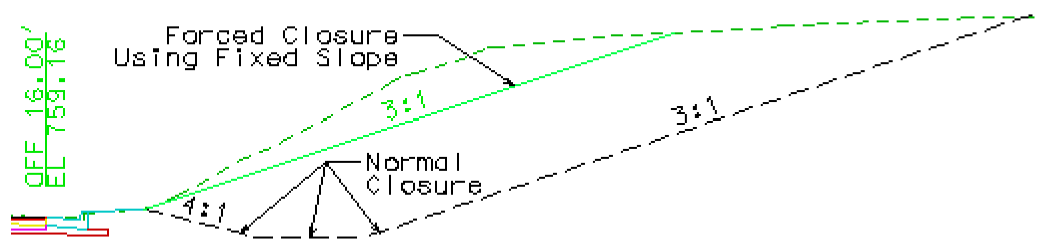


Cross-Section View Showing Force Closure Effect

Forced Closure using Redefinable Variables

If a force tie slope line is not used, **_s_LeftForceClosingSlopes** and **_s_RightForceClosingSlopes** can be use instead. To force the sloped closed set the variable for that side to ^Yes^. If normal closure is desired, set the variable to ^No^. The carets "^" are required.

The closure slope is controlled by redefinable variables using the Run:Rise format with the colon required. If the section is in cut Rise is ALWAYS positive and the variables **_s_LeftForceCutSlope** and **_s_RightForceCutSlope** control the slope for each side. If the section is in fill, Rise is ALWAYS a negative number (Example, 2:-1) and the closure slope for each side is controlled by **_s_LeftForceFillSlope** and **_s_RightForceFillSlope**.



The plan element will **ALWAYS** "override" these variables.

Cross-Section View Showing Force Closure Effect

Appendix 8 Special Ditches (Using Profiles to Set Ditch Elevation)

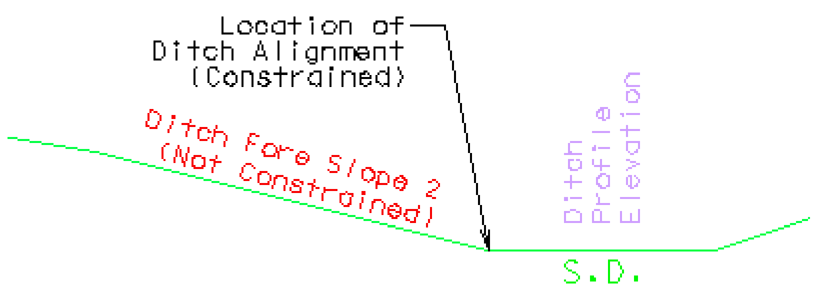
Drawn by Undivided New Pavement Typical Section.

Special Ditch Options

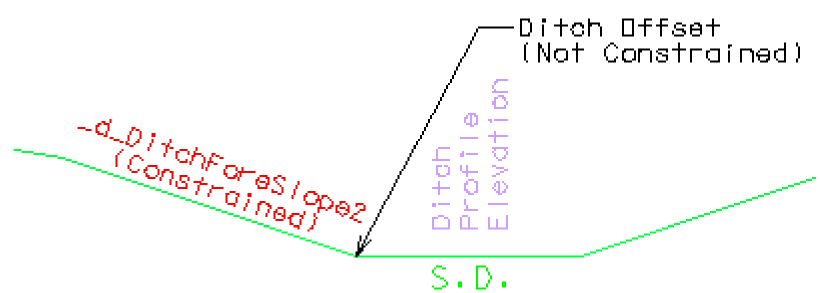
Special ditches allow the user to control the elevation of a ditch at any cross-section by defining a special ditch profile. Different profiles may be used for the left and right side. For the side slope to tie into a special ditch, either the offset to the ditch or the final ditch fore slope must unconstrained. Note: The use of a special ditch profile will **ALWAYS** "override" the standard ditch depth redefinable variables **_d_StandardDitchDepth_Left** and **_d_StandardDitchDepth_Right**.

If a ditch alignment is defined for the side slope condition, the offset to the ditch is fixed with the inside edge of the ditch starting at the ditch alignment and ditch fore slope 2 is unconstrained, consequently the redefinable variable for that side (**_d_DitchForeSlope2_Left** or **_d_DitchForeSlope2_Right**) is ignored. This condition is illustrated in the figure below and on the left.

If a ditch alignment is not defined for the side slope condition, ditch fore slope 2 for that side is fixed by **_d_DitchForeSlope2_Left** or **_d_DitchForeSlope2_Right** and the offset is unconstrained. This condition is illustrated in the figure below and on the right.



Special Ditch Location Constrained by Ditch Alignment
Fore Slope Before Special Ditch Is Unconstrained



Fore Slope Before Special Ditch Is Constrained
Offset to Special Ditch is Unconstrained

The above figures show the special ditches labeled "S.D." This lets the user know that a special ditch has been drawn of the this side slope condition. The user can specify this label using the redefinable variable **_s_SpecialDitchLabel**, Example: ^S.D.^. The carets "^" are required.

Special Ditch Profile and Alignment Variables

The variables for the special ditch COGO elements are in the **Define Variables** section of a proposed cross-section run. Theses elements must be stored in COGO before processing the cross section run. The station range for both the profiles and the ditch alignments must include the cross-section station for them to be used by the criteria. These and other features are discussed in more detail below.

Special Ditch Profiles

For a special ditch profile to be used it must be listed under the appropriate define variable and must be define for the cross-section station. Profiles for the left side slope condition must be listed in the value for the variable **Left Special Ditch Profiles** and profiles for the right side slope condition must be listed in the value for **Right Special Ditch Profiles**. Commas must separate the profile names. Examples: ltdit1,ltdit2,ltdit3 or rtdit1,rtdit2,rtdit3.

The criteria uses the following logic to determine the elevation for the special ditch: If the cross-section station is included in the station range for one of the special ditch profiles, the criteria will check to see if a ditch alignment is encountered. If the ditch alignment is found, the station where the pattern line crosses the ditch alignment is noted and the elevation from the profile at the ditch alignment station is used to set the ditch elevation. If a ditch alignment is not defined, the elevation from the profile at the cross-section station is used to set the ditch elevation. In other words, the elevation from the profile at the ditch alignment station is used if there is a valid ditch alignment for the side slope condition; otherwise, the elevation at the cross-section station is used.

Ditch Alignments

For a ditch alignment to be used it must be listed under the appropriate define variable. Only one ditch alignment per side per cross-section run is allowed. The alignment for the left side slope condition must be listed in the value for the variable **Left Ditch Alignment** (Example ltditch) and the one for the right side slope condition must be listed in the value for **Right Ditch Alignment** (Example rtditch).

Because the cross-section station must be included in the special ditch profile station range, it is recommended that the mainline station where the ditch alignment begins be used as the ditch alignment beginning station. If this is done, the station for the ditch alignment will be similar to the station for the cross-section.

If a ditch alignment is used, the elevation from the profile at the ditch alignment station is used to set the ditch elevation and the inside edge of the ditch will be at the location of the ditch alignment. This is helpful when trying to match an existing culvert for example. This will vary the ditch fore slope 2 variable and force the ditch to have varying ditch fore slopes. In other words, if a ditch alignment is used, the ditch fore slope redefinable variable is ignored.

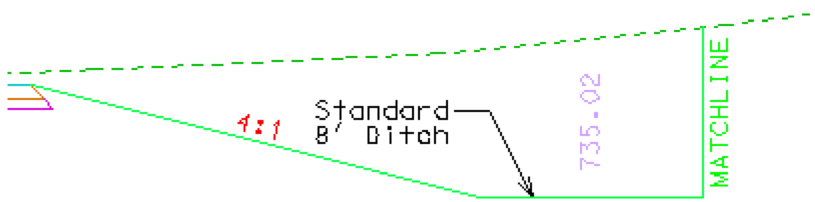
Appendix 9 Match Lines

Drawn by Undivided New Pavement Typical Section.

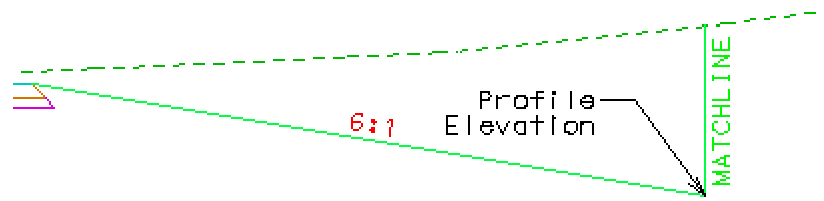
Match Line Options

Match lines allow the criteria to terminate a proposed ground side slope at a specific location and draw a vertical line from that location to existing ground, as shown in both of the figures below. This option is needed when the outside edge of a proposed side slope condition is adjacent to another proposed cross-section run, which occurs at interchanges and at any other situations where another proposed alignment runs along side the current run. Typically one of the alignments will control the elevation at the match line. The proposed cross-sections for the controlling alignment should be run first with a fixed final side slope. This is shown on the left below. Once the first set of cross sections is run, the Profile Grade Report may be used to create a profile and optional chain from these cross sections at the location of the match line. This profile and optional chain can be used to vary the slope of the proposed ground element by forcing it to end at the elevation obtained from the profile. This is shown on the right above.

There are two very different ways that the proposed ground surface can be modeled before the match line location is reached. Either a standard typical section can be used prior to the match line (as shown in the left figure below) or a straight line can be from the edge of shoulder, curb, or berm to a specific elevation at the location of the matchline, which is the method shown in the right figure below.



Typical Section Before Match Line



Single Element Before Match Line

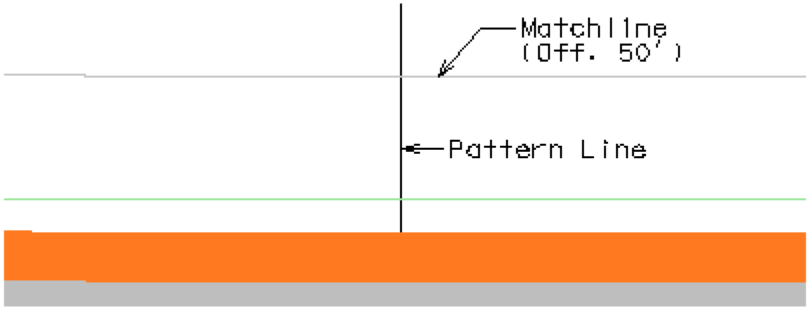
To draw a typical section prior to the match line (the case on the left above) a graphic element in the designated GEOPAK Lines DGN must be used is used to locate the match line. For this option, the redefinable variables described in the **Standard Side Slopes** Appendix are used to draw the proposed ground elements prior to the match line. The final proposed ground element mat be drawn either at a fixed or a variable slope. The variable slope for the final element is used if a specific elevation is desired at the match line.

The easiest way to have a single proposed ground element before the match line is to use a match line chain to indicate the location of the match line and a match line profile to determine the elevation desired.

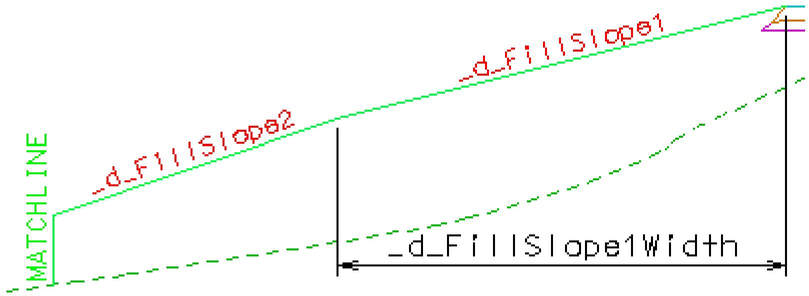
Each of these cases is treated separately below.

Typical Section Before Match Line Details

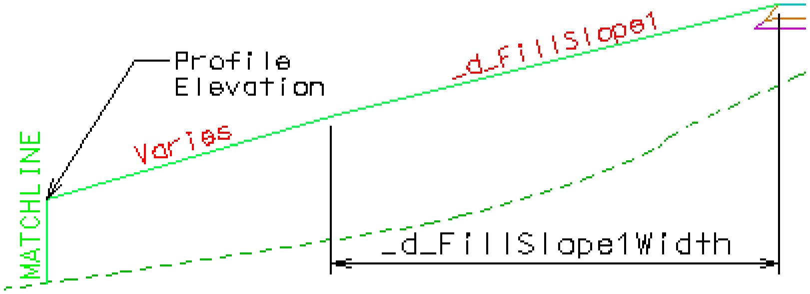
For any of the **Standard Side Slope** redefinable variables to be used by the criteria for this option, a graphic element is needed in the designated GEOPAK Lines DGN to indicate the location of the match line. This line must be drawn using the **D&C Manager** item **Matchline**, which is located in the D&C Manager path: "**Design Standards/Roadway/**". An example is shown in the figure to the right.



The slope on the final proposed ground element before the match line may be either a fixed value (as shown below in the figure on the left) or it can be variable with the element ending at an elevation obtained from a profile (as shown below in the figure on the right). To use a fixed slope for the final element, do NOT define a match line profile for that cross-section. If a match line profile is defined for the cross-section station, the final proposed ground element is drawn to the profile elevation. The match line profiles are defined for each side by using the Define Variables **Left Match Line Profiles** and **Right Match Line Profiles**. Commas must separate the profile names. Example: "ml1lt,ml2lt,ml3lt"; or "ml1rt,ml2rt,ml3rt". **Note:** The stationing of the profile must match the cross-section stationing.



Fixed Slope Option Used for Vertex at the Match Line
A Match Line Profile Is NOT Defined for the XS Station



Profile Used for Vertex at the Match Line
A Mach Line Profile Is Defined for the XS Station

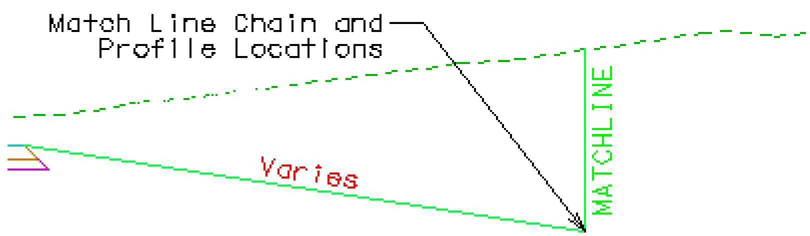
While the above examples are in fill, the same process works in cut. The only difference is the redefinable variables used.

Match Line Chain and Profile Details

If a single proposed ground element is to be drawn from the edge of shoulder, curb, or berm to a specified elevation at the match line, the use a match line chain to locate where the match line profile elevation is to be applied is the best option. The chain and profile can be easily obtained using the Profile Grade Report. **Note:** The use of a match line chain is not required; it is just the easiest option. To use the typical section option, the appropriate slope width redefinable variable would need to be large enough to extend past the match line location.

The Define Variables **Left Match Line Profiles**, **Right Match Line Profiles** and **Matchline Chain Names** list the profiles and chains to be used with match lines. The match line chain names must exactly match the corresponding profile name. Commas must separate the profile and the chain names. Example: “ml1lt,ml2lt,ml3lt”; or “ml1rt,ml2rt,ml3rt”. Up to eight matchline chain names are permissible per cross-section run. **IMPORTANT:** The list of the match line chains MUST NOT include any blank spaces. The stationing of the match line chain and corresponding profile must be the same. Also, the presence of a defined match line chain will **ALWAYS** "override" a graphic match line.

The Redefinable Variable **_d_MatchLineChainSearchDistance** sets the distance to look for a match line chain. The distance is measured in master units from the edge of shoulder, curb, or berm. This is a positive value and MUST be greater than zero.



Example Using a Match Line Chain and Profile

Appendix 10 Rock Benches

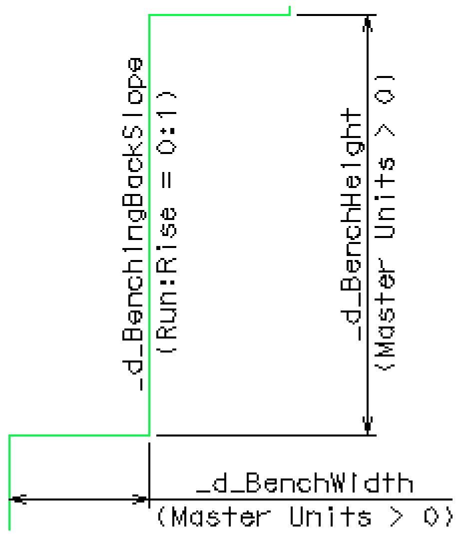
Drawn by Undivided New Pavement Typical Section.

Cross-Section Geometry Required

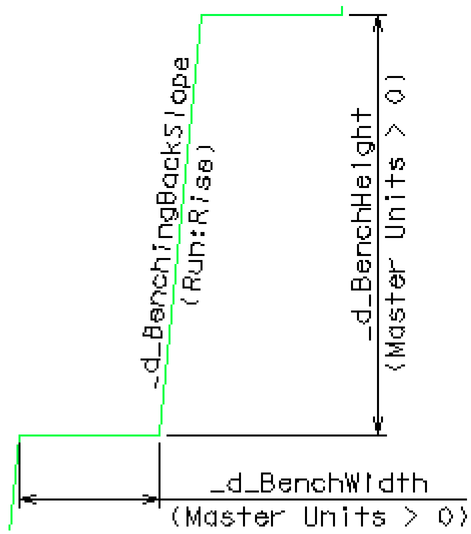
If the proposed ground is below a top of rock line in the cross-section, rock benches will be drawn starting at the back of the ditch. For this to work, the top of rock line must be drawn in the designated Cross Section DGN using symbology matching the search parameters for the **D&C Manager** item **Rock Top**, which is located in the path: "**Design Standards/Cross Sections/Existing Surfaces/**". The current search symbology is limited to Level 58, Color 8, Weight 3, and Style 2, which must be matched exactly.

Rock Bench Redefinable Variables

Four redefinable variables control how the rock benches are drawn. The bench width and the bench height are set by **_d_BenchWidth** and **_d_BenchHeight**, respectively, in master units, which must be greater than zero. The rock face can either be vertical or drawn at an inclined slope, which is set by **_d_BenchBackSlope**, in Run:Rise format. Rise is ALWAYS a positive number. For a vertical face set _d_BenchBackSlope to 0:1. For an inclined face the Run value is the percent of slope from vertical divided by 100. Consequently, to have a 10% slope to the incline, use a value of 0.1:1, as shown in the following figures

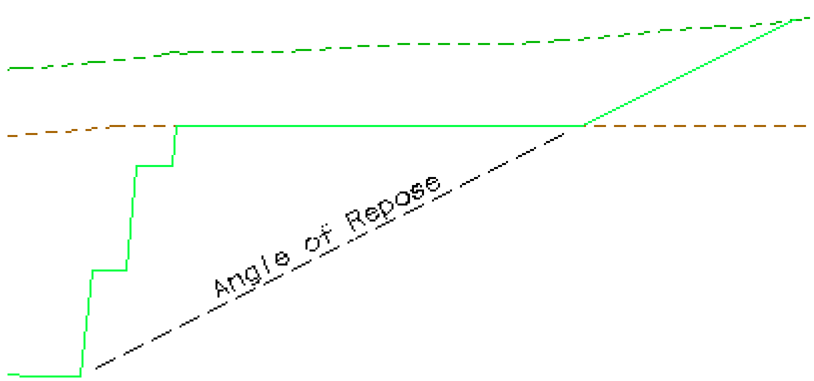


Vertical Rock Face (_d_BenchBackSlope = 0:1)

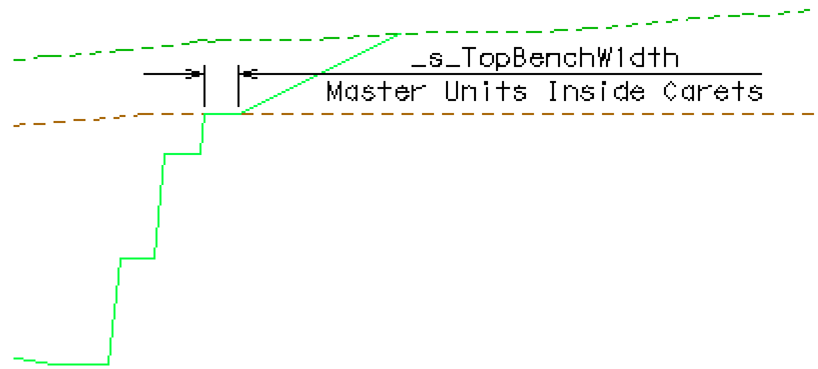


Inclined Rock Face (_d_BenchBackSlope = Run:Rise)

The fourth rock bench redefinable variable is **_s_TopBenchWidth**, which locates the toe of the slope for the Class A material above the rock. Two options are available. The toe can be located either at the angle of repose, which is the angle of the ditch back slope projected from the back of the ditch as shown in the figure on the left below, or at a specified horizontal distance (in master units) from the rock face as depicted in the figure on the right. Set _s_TopBenchWidth = ^RS^ for angle of repose slope or put the horizontal distance between carets (example ^10^) to specify an actual width. The carets "^" are required. **Note**: For the slope to be drawn correctly, the existing ground line must extend far enough for the proposed ground line to intersect it. Inconsistent results will occur if the existing ground line is not wide enough.



Example: _s_TopBenchWidth = ^RS^
Toe of Slope at Angle of Repose Slope



Example: _s_TopBenchWidth = ^10^
Toe of Slope at 10’ From Rock Face

Appendix 11 Sidewalk

Drawn by Undivided New Pavement Typical Section.

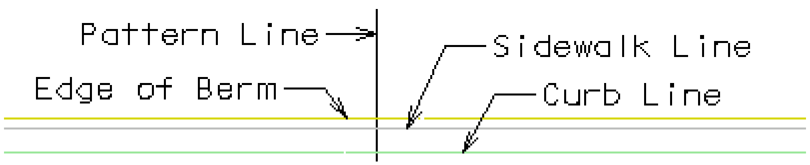
Plan View Geometry

Two conditions need to be met before sidewalk will be drawn in a cross-section. The first is that the edge of sidewalk is drawn in the designated Proposed Plan DGN. The line must be drawn using the **D&C Manager** item **Sidewalk** located in the D&C Manager path: "**Design Standards/Roadside/**". The second condition is that the sidewalk is located within the side slope **berm**.

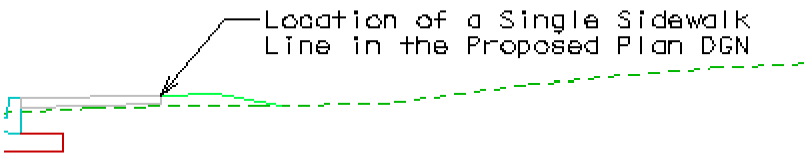
If the sidewalk is adjacent to the back of a curb, only one edge of sidewalk line away from the back of the curb is to be drawn. However, if the sidewalk is not adjacent to a back of curb, both edge of sidewalk lines need to be present. The sidewalk will not be drawn if the criteria cannot match up an edge of sidewalk line with either a curb line or another edge of sidewalk line. The required lines and the results are given in the following examples (**Note**: The width of the sidewalk is determined by the plan view geometry):

Example 1: Sidewalk Adjacent to Back of Curb

The sidewalk is drawn between the back of curb and a single sidewalk line:



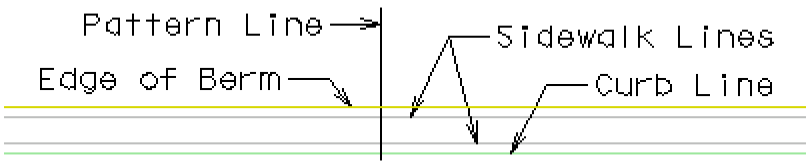
Plan View Geometry With One Sidewalk Line



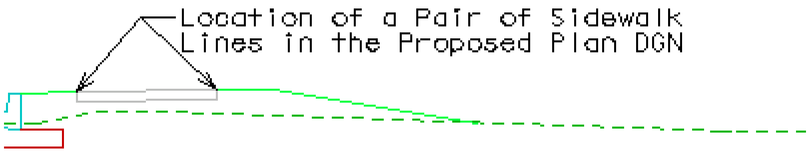
Cross-Section View

Example 2: Sidewalk Away from Back of Curb

The sidewalk is drawn between two matching sidewalk lines:



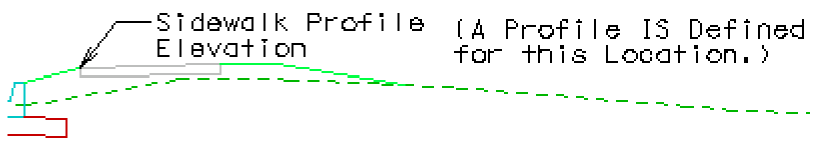
Plan View Geometry With Two Sidewalk Lines



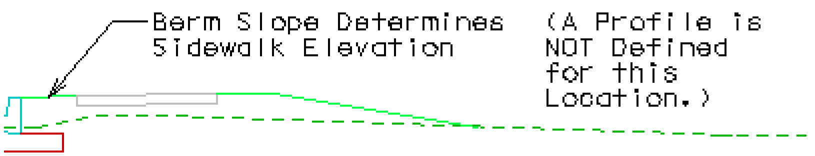
Cross-Section View

Sidewalk Elevation

The elevation of the sidewalk in the cross-section view is set by the elevation assigned to the inside edge of the sidewalk. This elevation can be determined either by the slope of berm or by a previously defined profile. The Define Variables **Left Sidewalk Profiles** and **Right Sidewalk Profiles** list the profiles to be used with a sidewalk for each side of the cross-section. If a sidewalk profile is defined for the cross-section station and side, the inside edge of the sidewalk is started at the profile elevation for that station, as shown in the figure on the left below. If a sidewalk profile is NOT defined for the cross-section station and side, elevation of the inside edge of the sidewalk is set by the cross-section geometry. If the sidewalk starts at the back of a curb, the inside edge of the sidewalk is drawn at the elevation of the back of the curb. If the sidewalk is drawn between to sidewalk lines, the berm slope is projected to the inside sidewalk edge, which is the starting elevation for the sidewalk, as shown below in the figure on the right.



Cross-Section With a Sidewalk Profile Defined

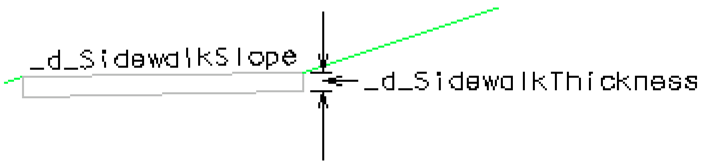


Cross-Section Without a Sidewalk Profile Defined

The elevation of the outside edge of the sidewalk is controlled by the redefinable variable **_d_SidewalkSlope**, which sets the proposed sidewalk cross slope in percent format from the inside edge to the outside edge, as shown in the figure below. Do not include the percent sign. Do include the negative sign when applicable.

Sidewalk Thickness

The redefinable variable **_d_SidewalkThickness** sets the sidewalk thickness in master units, as shown in the following figure. This is a positive value and MUST be greater than zero. **Note**: The width of the sidewalk is determined by the plan view geometry.



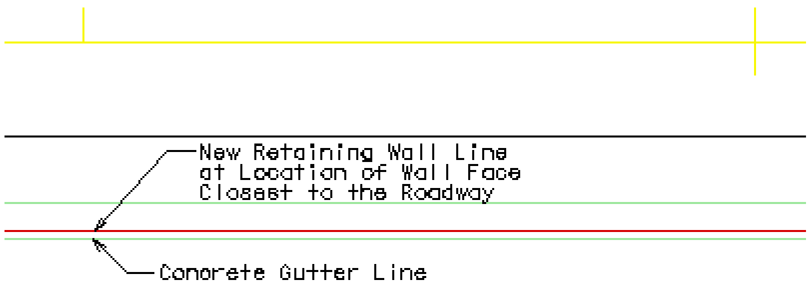
Sidewalk Dimension Redefinable Variables

Appendix 12 Retaining Walls

Drawn by Undivided New Pavement Typical Section.

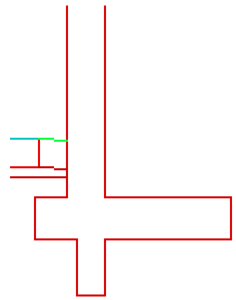
Plan View Geometry

Retaining walls are located in the proposed side slope by placing a new retaining wall line in the designated Proposed Plan DGN at the location of the face of the wall closest to the roadway. The line must be drawn using the **D&C Manager** item **New Retaining Wall** located in the D&C Manager path: "**Design Standards/Safety and Structures/**", as shown in the figure to the right. The retaining wall may be located either adjacent to a shoulder or proposed ground. A gutter can be place behind the wall by locating Concrete Cutter line at the backside of the wall in cut (D&C Manager path: "**Payitems/Roadway/Curb/**").



Proposed Surfaces Adjacent to the Wall

If the retaining wall is to be placed adjacent to the shoulder, two conditions must be met for the cross-section to be drawn correctly. The first is that there must be a gap between the retaining wall and the shoulder as drawn in the cross-section. This is required for the criteria to recognize the retaining wall line in the plan view. Typically, joint filler is placed between the retaining wall and the shoulder and the width of the gap can be used to represent the joint filler thickness. For non-earth shoulders, the second condition is that the shoulder must be recognized by the criteria as concrete regardless of the actual material used. This is required so that the shoulder edge of the shoulder layers are drawn vertically nest to the wall. If the edge of the shoulder is within the distance in master units set for the redefinable variable **_d_AdjacentWallSearchDistance**, the aggregate layer will be extended to the wall as shown in the figure to the right. Use a positive number.



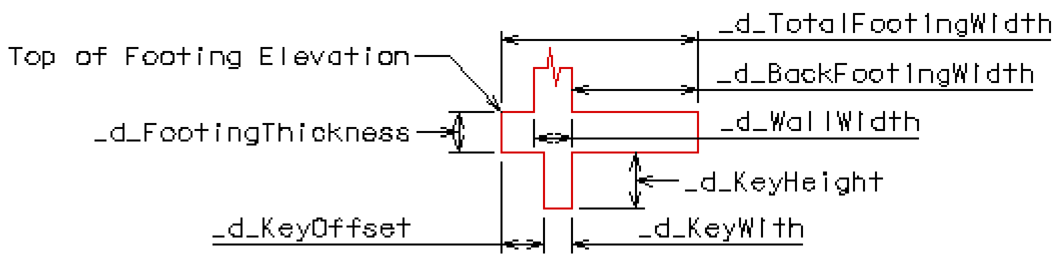
If proposed ground is adjacent to the retaining wall, a maximum of two different slopes are permitted before the retaining wall. The value of these slopes are the slope assigned to the berm (if one is drawn) and to the first fore slope after the berm that is actually drawn. The fore slope that is applied depends on whether the side slope is in cut or fill and the side slope parameters applied to the cross-section.

Footing Cross-Section Details

The dimensions for drawing the retaining wall footing are set by the redefinable variables **_d_WallWidth** (measured from the retaining wall line in the plan dgn in a direction away from the roadway), **_d_BackFootingWidth** (measured from the edge of the wall with the deepest fill to the back or heel of the footing), **_d_TotalFootingWidth** (measured from the back or heel of the footing to the front or toe of the footing), **_d_FootingThickness** (measured down from the top of the footing), **_d_KeyHeight** (measured down from the bottom of the footing), **_d_KeyOffset** (measured from the front or toe of the footing to the front edge of the key), and **_d_KeyWidth** (measured from the front of the key to the back of the key), as shown in the following figure. All values are in master units and may be zero or greater than zero. The back footing width will always be drawn on the side of the wall with the deepest fill above the footing.

The elevation of the top of the footing can be set by either defining a top of footing profile or by a minimum depth below ground. To specify a top of footing elevation, create a profile based on the cross-section station. Different profiles may be used for the left and right side and are listed using the define variables **Left Top of Footing Profiles** and **Right Top of Footing Profiles** respectively. These profiles must be previously stored in COGO before processing cross sections. The profile names are to be separated by commas. Example: LTF1,LTF2,LTF3 or RTF1,RTF2,RTF3. If the of footing profile exists for the cross-section station, the elevation for that station will be used as the top of footing elevation.

If a top of footing profile does not exist for the station, the top of footing will be drawn at a specified depth below either the proposed or existing ground line, whichever is lower. If the proposed ground is lower, the wall is in cut and the top of footing will be the distance specified by the redefinable variable **_d_DepthBelowProposedGroundInCut**. However, if existing ground is lower, the wall is in fill and the top of footing will be the distance specified by the redefinable variable **_d_DepthBelowExistGroundInFill**. Both values are in positive master units.



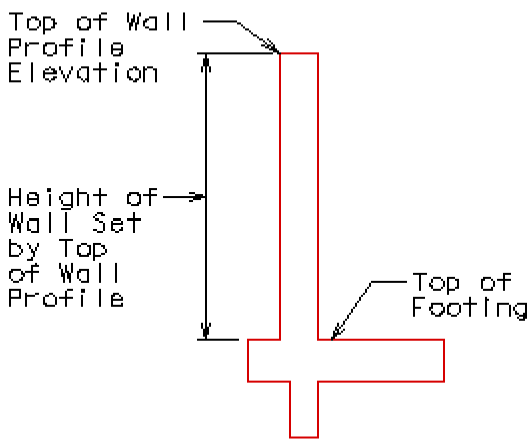
Retaining Wall Footing Dimensions

Retaining Wall Height

The height of the retaining wall may be set by specifying the elevation of the top of the wall, by defining a specific height, or by letting it vary with the ground surface. A profile is used to set a specific elevation for the top of the wall based on the cross-section station. If a top of wall profile does not exist for the cross-section station, the redefinable variable **_s_WallHeight** controls the height of the wall. This variable can be set to specify a set value for the height of the wall or it can be set to allow the height of the wall to vary with ground surface.

Height Set by Elevation of the Top of the Wall

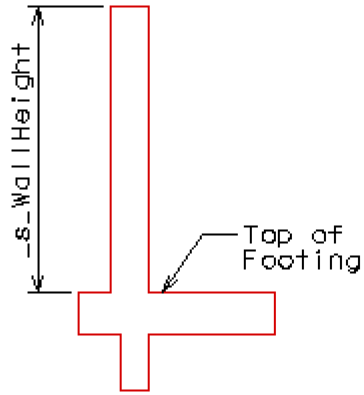
To set the wall height by specifying a top of wall elevation, create a profile based on the cross-section station. Different profiles may be used for the left and right side and are listed using the define variables **Left Top of Wall Profiles** and **Right Top of Wall Profiles** respectively. These profiles must be previously stored in COGO before processing the cross sections. The profile names in the list need to be separated by commas. Example: LTW1,LTW2,LTW3 or RTW1,RTW2,RTW3.



If the of top of wall profile exists for the cross-section station, the elevation for that station will be used as the top of wall elevation and the wall will be drawn between the top of the footing and the top of wall elevation, as shown in the figure to the right. The proposed ground settings will be used to connect the retaining wall to the existing and the proposed ground. The rest of the parameters for the wall and footing dimensions are the same as those shown on the previous page.

Height Set by a Fixed Value

The redefinable variable **_s_WallHeight** controls the height of the wall if a top of wall profile is not used. As a string redefinable variable (as defined by the beginning “_s_”), it is used control whether the height of the wall is a fixed height or variable. If a fixed height is desired, set the variable to a numeric value inside of carets (^). Example: Assign it the value of ^10^ for a 10 foot high wall. AWAYS use the carets.



The fixed height of the wall is measured up from the top of the footing in master units, as shown in the figure to the right. The proposed ground settings will be used to connect the retaining wall to the existing and the proposed ground. The rest of the parameters for the wall and footing dimensions are the same as those shown on the previous page.

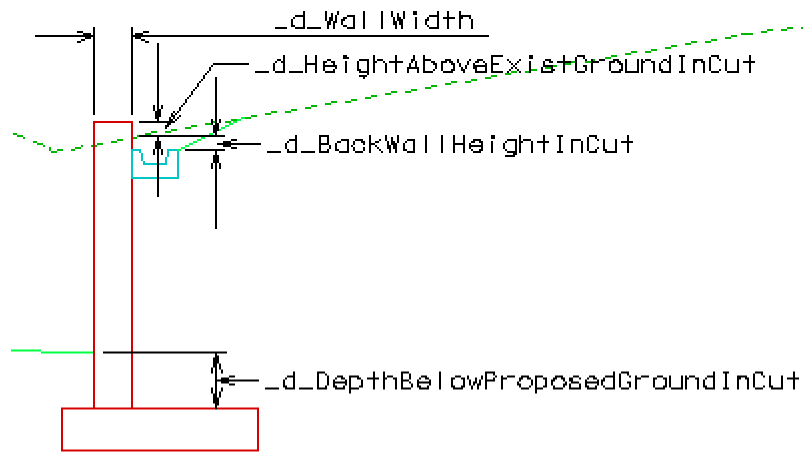
Home

Height Varies With the Ground Surface

To let the height of the wall vary with the ground surface, set _s_WallHeight to ^GS^ for ground surface. AWAYS use the carets. With the variable set to this value, different parameters are allowed for cut and fill. Each of these cases in considered separately.

Variable Wall Height in Cut

If _s_WallHeight is set to ^GS^ and the wall is in cut, the redefinable variable _d_HeightAboveExistGroundInCut controls the top of the wall by specifying the distance in master units that the top of the wall is above the existing ground on the back side of the wall, as shown in the figure to the right.

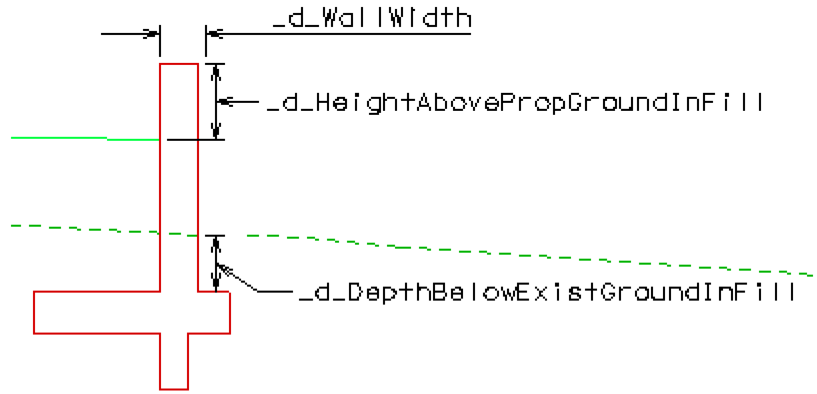


If a gutter is to be drawn on the backside of the retaining wall the redefinable variable _d_BackWallHeightInCut controls the distance in master units that the top the gutter is below existing ground, as shown to the right.

If a profile is not used to locate the top of the footing, the redefinable variable _d_DepthBelowProposedGroundInCut controls the depth of the footing below the proposed ground surface as indicated in the figure to the right. This distance is in master units.

Variable Wall Height in Fill

If _s_WallHeight is set to ^GS^ and the wall is in fill, the redefinable variable _d_HeightAbovePropGroundInFill controls the top of the wall by specifying the distance in master units that the top of the wall is above the proposed ground on the roadway side of the wall, as shown in the figure to the right.



If a profile is not used to locate the top of the footing and the wall is in fill, the redefinable variable _d_DepthBelowExistGroundInFill controls the depth of the footing in master units below existing ground as indicated in the figure to the right. Both sides of the wall are checked and the lower elevation is used.

The rest of the dimensions for the retaining wall are the same as those used in the section dealing with the retaining wall footing.